

SAI-82-738-HU  
009/53(241)

REEDS COMPUTER CODE

FINAL REPORT

(NASA-CR-161956) REEDS COMPUTER CODE Final  
Report (Science Applications, Inc.,  
Huntsville, Ala.) 219 p HC A10/MF A01  
CSCI 09B

N82-16760

Unclas  
G3/61 07990

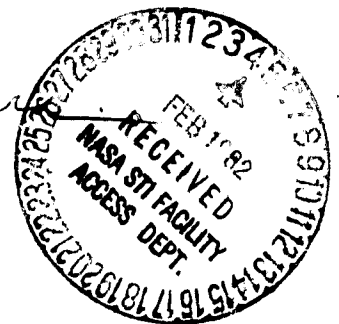
23 DECEMBER 1981

*NAS 8-31806*

PREPARED BY:

CHARLES BJORK

*Science Applications, Inc.  
2109 W. Clinton Ave.  
Huntsville, Alabama*



**SAI**

## TABLE OF CONTENTS

<u>SECTION</u>		<u>PAGE</u>
1.0	INTRODUCTION.....	1-1
2.0	REEDS MODEL DESCRIPTION AND BASIC STRUCTURE.....	2-1
2.1	MODULE DESCRIPTION.....	2-1
2.1.1	The REEDS Executive Module.....	2-15
2.1.2	The RCLDS Cloud Rise Module.....	2-22
2.1.3	The RMTPS Meteorological Data Plotting Module....	2-39
2.1.4	The RCONS Centerline Concentration Module.....	2-41
2.1.5	The RISOS Isopleth Plotting Module.....	2-45
3.0	REEDS OPERATION.....	3-1
3.1	THE TEST CASE.....	3-1
4.0	LISTINGS.....	4-1
4.1	MAIN PROGRAM MODULES AND SUBROUTINE PACKAGES.....	4-2
4.2	AUXILIARY SUBROUTINE LISTINGS.....	4-123
4.3	MAIN MODULE QUESTION FILES.....	4-137
4.4	EXAMPLE METEOROLOGICAL SOUNDING FILES .....	4-142



## LIST OF ILLUSTRATIONS

<u>FIGURE</u>		<u>PAGE</u>
1-1	REEDS Program Segment Scheduling Hierarchy.....	1-3
2-1	Illustration of the "Thermodynamic" Phase Modeled in REEDS....	2-2
2-2	Illustration of the "Kinetic" Phase Modeled in REEDS.....	2-3
2-3	Structured Diagram of the REEDS Code Showing Important and Commented Operations.....	2-4
2-4	Program Excerpt Showing the Common Data Arranged in "Pseudo" Blocks.....	2-8
2-5	Detailed Structure Flow Diagram of Subroutine RSGAZ.....	2-29
3-1	Code Initiation.....	3-3
3-2	Configurational Paths Added.....	3-3
3-3	Launch Specific Data Added.....	3-4
3-4	Initial Printer Data-Header and Meteorological Data.....	3-5
3-5	Blank Meteorological Form.....	3-7
3-6	Meteorological and Initial Cloud Data Plotted with Estimated Mixing Layer Height.....	3-8
3-7	Modifying the Mixing Layer Height.....	3-9
3-8	Modified Cloud - Showing Mixing Layer Height Effects.....	3-10
3-9	Final Exhaust Cloud Rise Data.....	3-11
3-10	Centerline Hcl Concentrations/Dosages with Cloud Arrival/ Departure Times and Pad-Relative Positions.....	3-12
3-11	Centerline Concentration/Dosage Plots Added.....	3-13
3-12	Blank Form for Centerline Concentration/Dosage Plots.....	3-14
3-13	Centerline Dosage/Concentration Plots.....	3-15
3-14	Addition of Maximum Points of Concentrations and Dpsage Along Ground Track.....	3-16
3-15	Blank Sea Map.....	3-18
3-16	Blank Land Map.....	3-19
3-17	Specific Site Range/Azimuth Entry.....	3-20
3-18	Off-Centerline Concentrations and Dosages with $\pm 10^0$ Uncertainties for all Effluents.....	3-21
3-19	Off-Centerline Point Plotting.....	3-22
3-20	Default Isopleth Values, Isopleth Plotting, and Program Termination.....	3-23
3-21	Isopleth Plotting.....	3-24
3-22	Cloud Position and Extent as a Function of Time.....	3-25

## LIST OF TABLES

<u>TABLE</u>		<u>PAGE</u>
2-1	COMMON DATA DESCRIPTION.....	2-9



## 1.0 INTRODUCTION

The REEDS computer code is a tool useful for the estimation of certain rocket exhaust effluent concentrations and dosages and their distributions near the earth's surface following a rocket launch event. Output from REEDS has been useful in producing near-real-time air quality and environmental assessments of the effects of certain potentially harmful effluents, namely HCL,  $Al_2O_3$ , CO, and NO. REEDS is the last in an evolutionary series of codes which treats the atmosphere within the first few kilometers of the earth's surface as a single layer; thus the name REEDS - Rocket Exhaust Effluent Diffusion Single layer model.

REEDS was developed for the NASA tropospheric environmental program by the NASA/Marshall Space Flight Center (MSFC) Environmental Effects Task Team (EETT) member, Science Applications, Inc. NASA leadership was provided by the EETT leader, Dr. J. Briscoe Stephens. As a result of continuing research, the function of REEDS is being largely replaced by the more comprehensive multi-layer diffusion program, REEDM. Therefore, this document should valuably assist the process of evaluating the progress made by the application of new techniques to the assessment problem.

REEDS is designed to operate on a Hewlett Packard 1000 series computer system. Research, development and testing took place on the MSFC system at the Space Sciences Laboratory (SSL), followed by implementation and subsequent operational monitoring support and bio-medical studies via the system at the Kennedy Space Center (KSC). A modest effort was made to keep these systems compatible for modeling software operations purposes, even though their diverse purposes obviously gave the systems different capabilities, peripheral hardware, and software.

An important feature of the HP1000 series computers used in the REEDS code is the Program Scheduling feature. The feature permits one self-contained program to initiate another completely self-contained program, and then return operation to the first. This is somewhat analogous to the FORTRAN feature of calling SUBROUTINES. Thus, a program which completely fills the available



program memory may execute another, which may execute another, etc., greatly expanding the capabilities of the computer system. Thus, the main executive module schedules major modeling modules, which, in turn schedule other modules, facilitating a logical division of research and modeling effort. The resulting overall structure in the REEDS application is illustrated in Figure 1-1.

REEDS was originally developed for interactive graphical operation through an SAI Plasmascope<sup>TM</sup>\*. The requirement for compatibility with the KSC installation necessitated the development of a graphics handler permitting operation with or without the Plasmascope<sup>TM</sup>. This handler uses several assembly-level programmed subroutines, which constitute the only non-FORTRAN software in the package, and which permit complete operation via the HP Cathode Ray Tube (CRT) terminals as well as the Plasmascope<sup>TM</sup>.

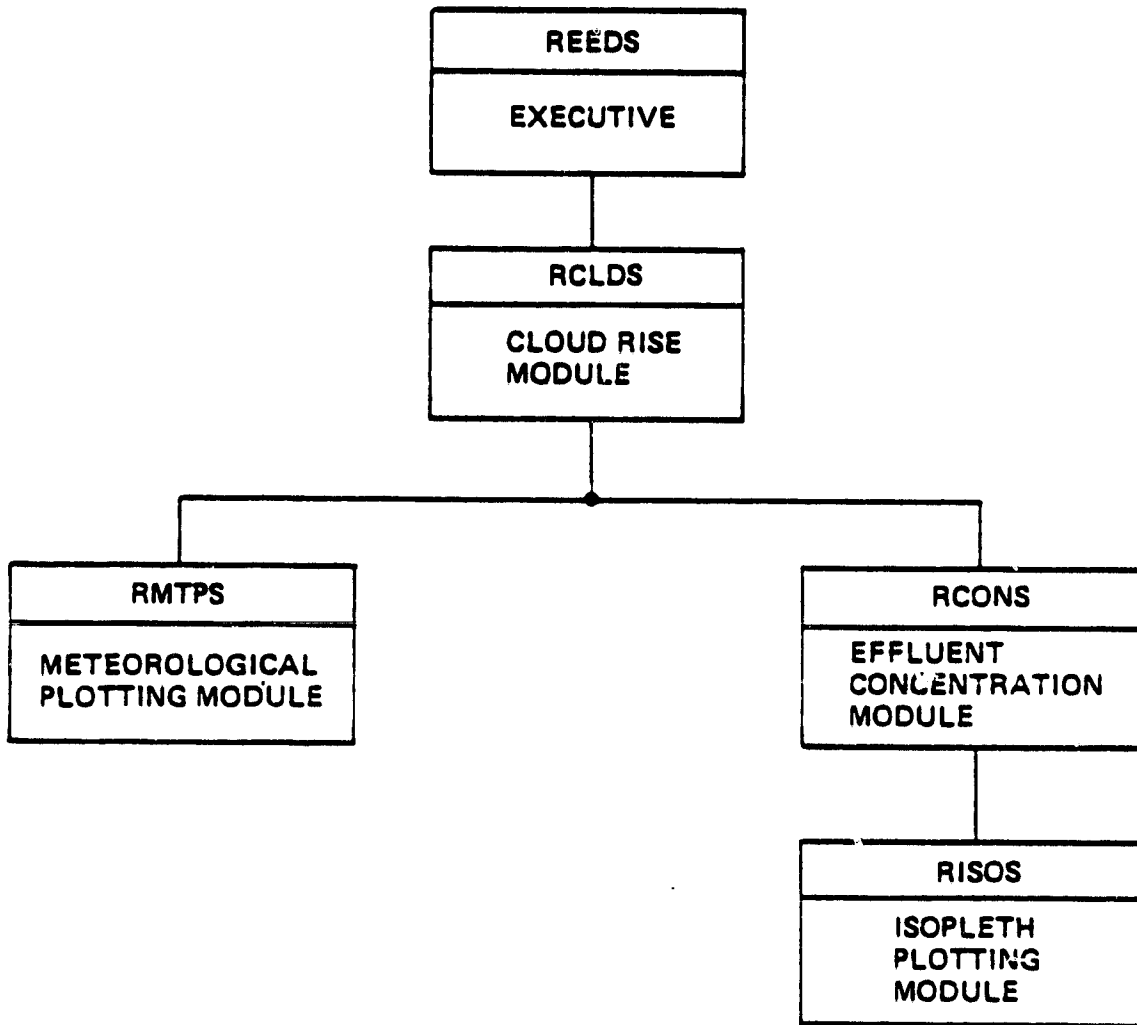
While the REEDS code is set up primarily for operation on Titan, Delta, and Space Shuttle Vehicles at their respective KSC/Cape Canaveral Air Force Station (CCAFS) sites, it has a capability for far more general applications. The vehicle data, primarily source rate and altitude histories, are input in parametric form. Simple replacement of these and the site coordinates allows the user to apply the code to a far more general class of problems. This is especially true since the code can handle both the continuous and instantaneous sources.

Further documentation regarding structure and modeling will be found in Section 2 of this document. Operational guidance is contained in Section 3, and program and input data file listings will be found in Section 4.

---

\*Trademark of an SAI Technology Division Product.





SAI-12817

FIGURE 1-1. REEDS PROGRAM SEGMENT SCHEDULING HIERARCHY



## SECTION 2.0

### REEDS MODEL DESCRIPTION AND BASIC STRUCTURE

The overall structure of the REEDS code is broken down into blocks which largely mirror the phenomenological categorization of the effluent diffusion processes. There is first a "Thermodynamic Phase" in which a rocket exhaust cloud is generated, transported and expanded until approximate thermodynamic equilibrium with its environment is reached (Figure 2-1). The RCLDS module (cloud rise module) contains most of the modeling for this phase. RCLDS calls RMTPS, which illustrates the local meteorology on a plotter for the user, showing approximate geometric relationships with respect to the effective atmospheric boundary layer interface. Then, thermal equilibrium having been reached, the turbulent diffusion process is allowed to proceed (Figure 2-2) in the "Kinetic Phase". The RCONS module contains most of the modeling for this process, and plots the concentration and dosage as a function of range along the cloud centerline. RCONS also calls RISOS, which plots concentration and dosage contours (called "ISOPLETHS") on a map of the launch vicinity. Of course, control, meteorological, and configuration input is provided by the main REEDS executive module, as is sequence control.

To make input more flexible, use is made of "question" input files. For each module for which it is necessary, user input prompts are stored in an indexed file, from which a module may draw in order to prompt the user for a particular input. Changes and/or additions may then be made as necessary to these files. Listings of these files are shown in Section 4.

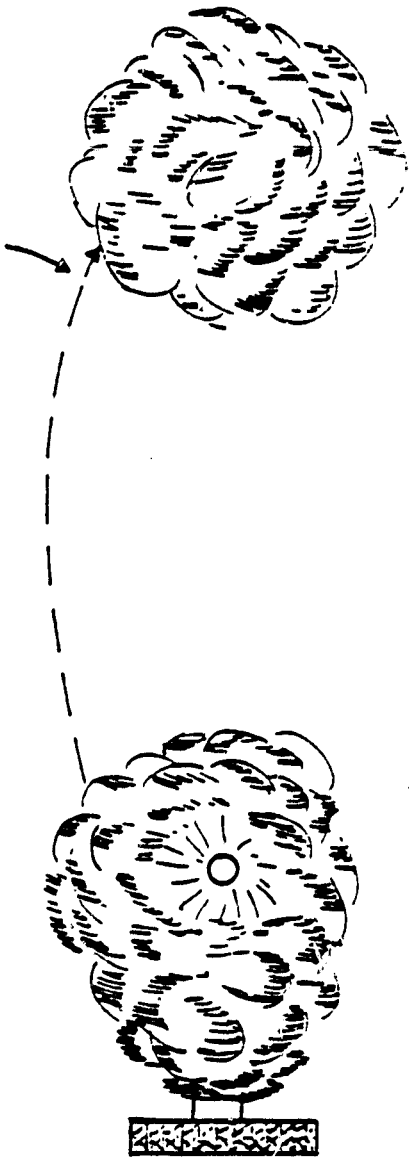
The structured diagram of Figure 2-3 illustrates the gross operational sequence of the entire code. This will be elaborated upon in the following subsections.

#### 2.1 MODULE DESCRIPTION

Each module will now be roughly described in narrative fashion, seasoned with the appropriate mathematical expressions employed in the modules. As much as possible, the narrative "walk" through each module will follow the commenting in the code. Some commenting not directly related to calculational procedures may be excluded from the description. In addition to making use of

ORIGINAL PAGE IS  
OF POOR QUALITY

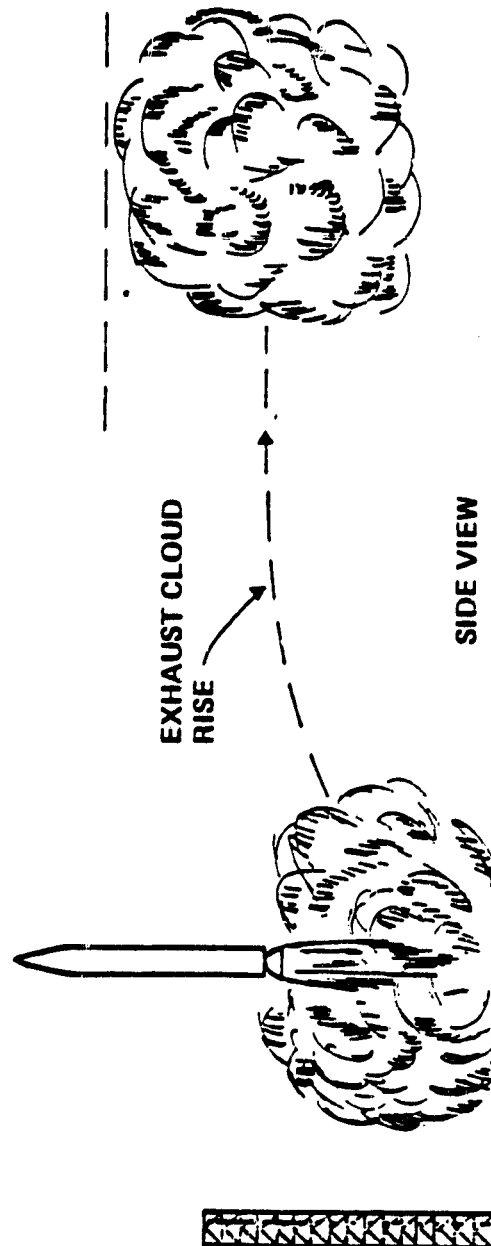
EXHAUST CLOUD GROUND TRACK



TOP VIEW

TOP OF ATMOSPHERIC  
MIXING LAYER

POINT OF THERMAL  
EQUILIBRIUM



SIDE VIEW

EXHAUST  
CLOUD  
GENERATION

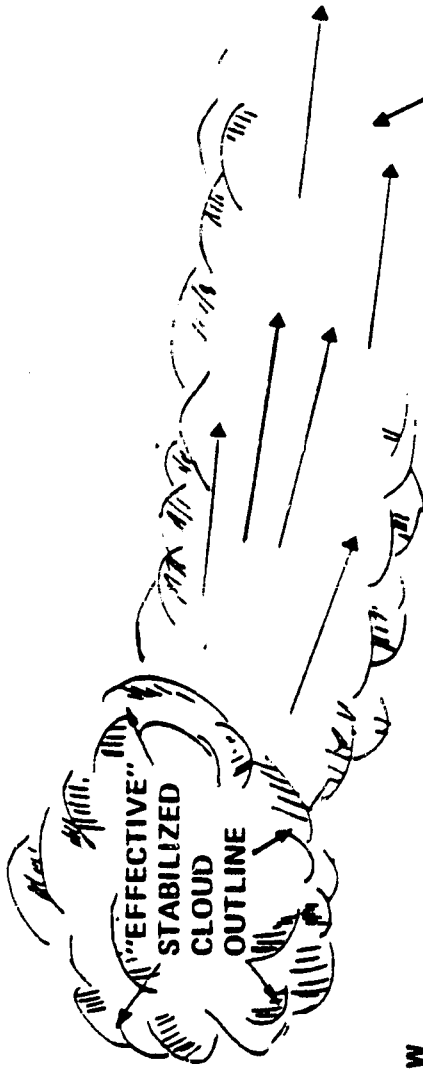
EXHAUST  
CLOUD  
RISE

EXHAUST CLOUD  
STABILIZATION

SAI-12621

FIGURE 1. ILLUSTRATION OF THE THERMODYNAMIC PHASE MODEL FOR IN RISES

WIND FIELD

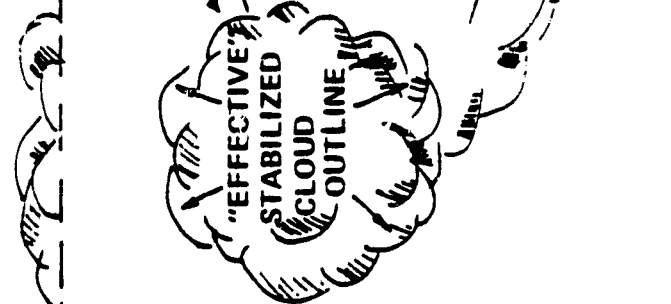


TOP VIEW

PORTION OF CLOUD DENIED TO THE KINETIC PHASE BY LOCATION OF TOP OF MIXING LAYER

TURBULENT DIFFUSION

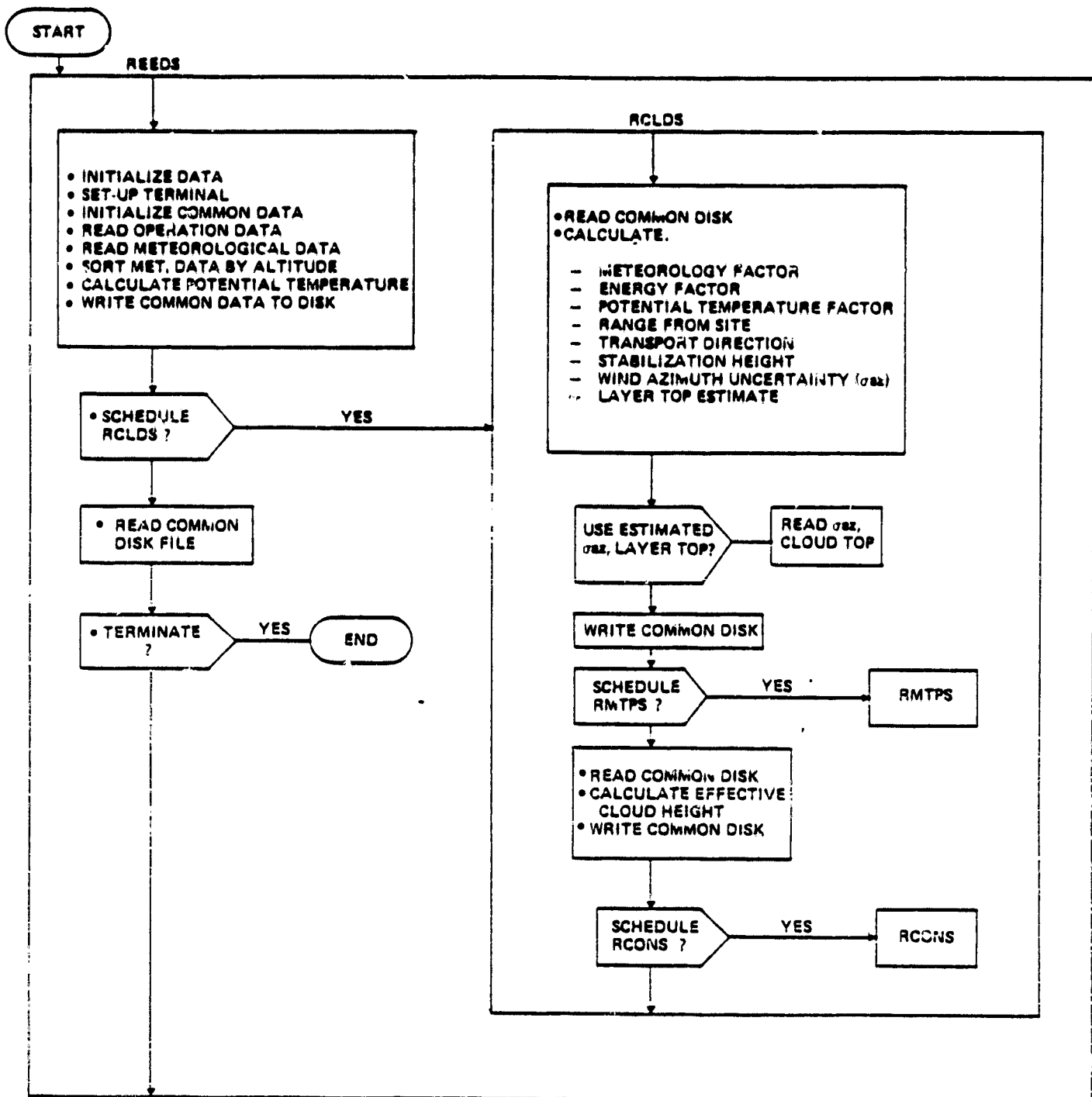
RESULTING "EFFECTIVE" CLOUD AFTER DELETION OF PORTION EXTERNAL TO MIXING LAYER AND SUBSEQUENT DIMENSIONAL ADJUSTMENT



SIDE VIEW

WIND FIELD

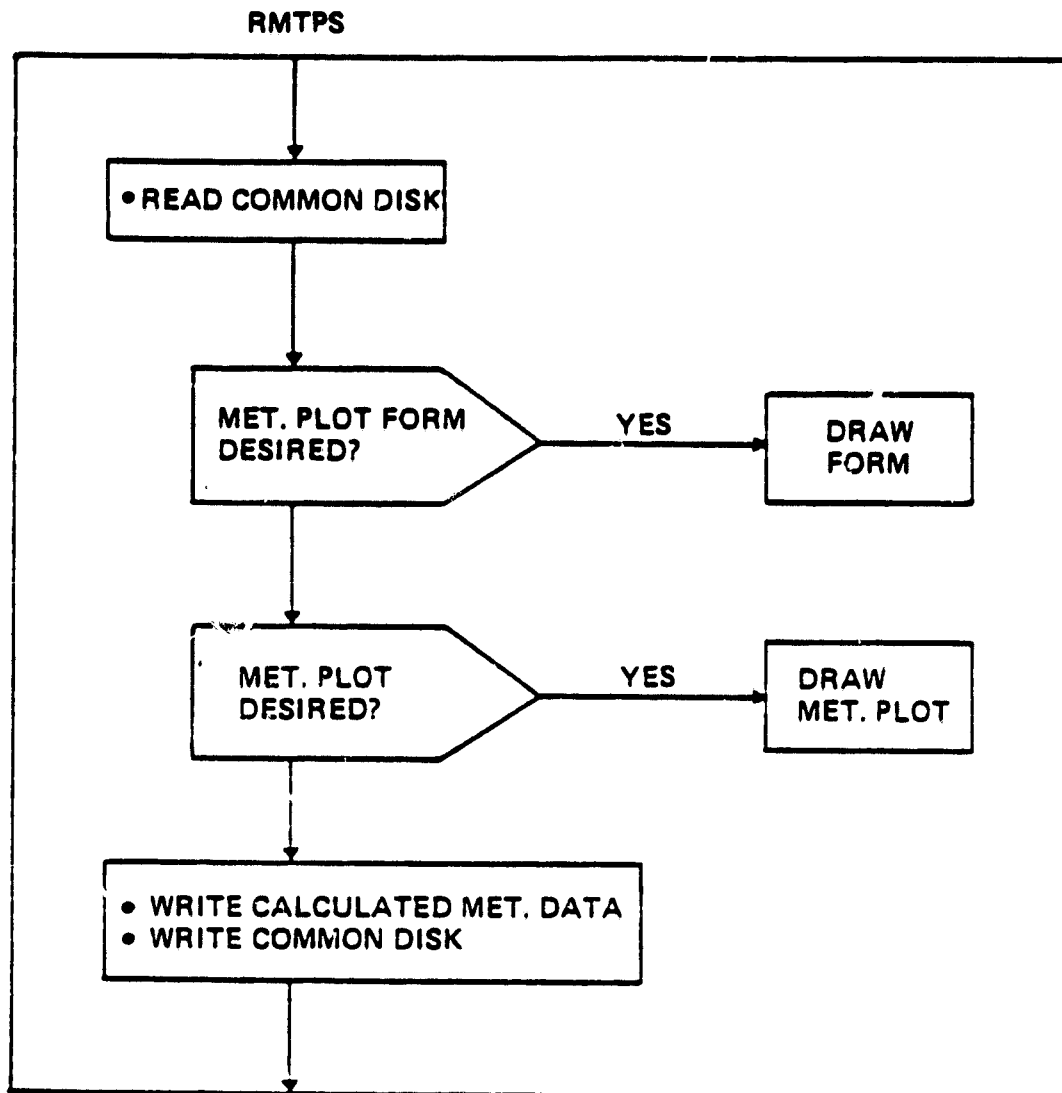
FIGURE 2.2. ILLUSTRATION OF THE "KINETIC" PHASE MODELED IN REEDS



SAI-12814

FIGURE 2-3. STRUCTURED DIAGRAM OF THE REEDS CODE SHOWING IMPORTANT AND COMMENTED OPERATIONS

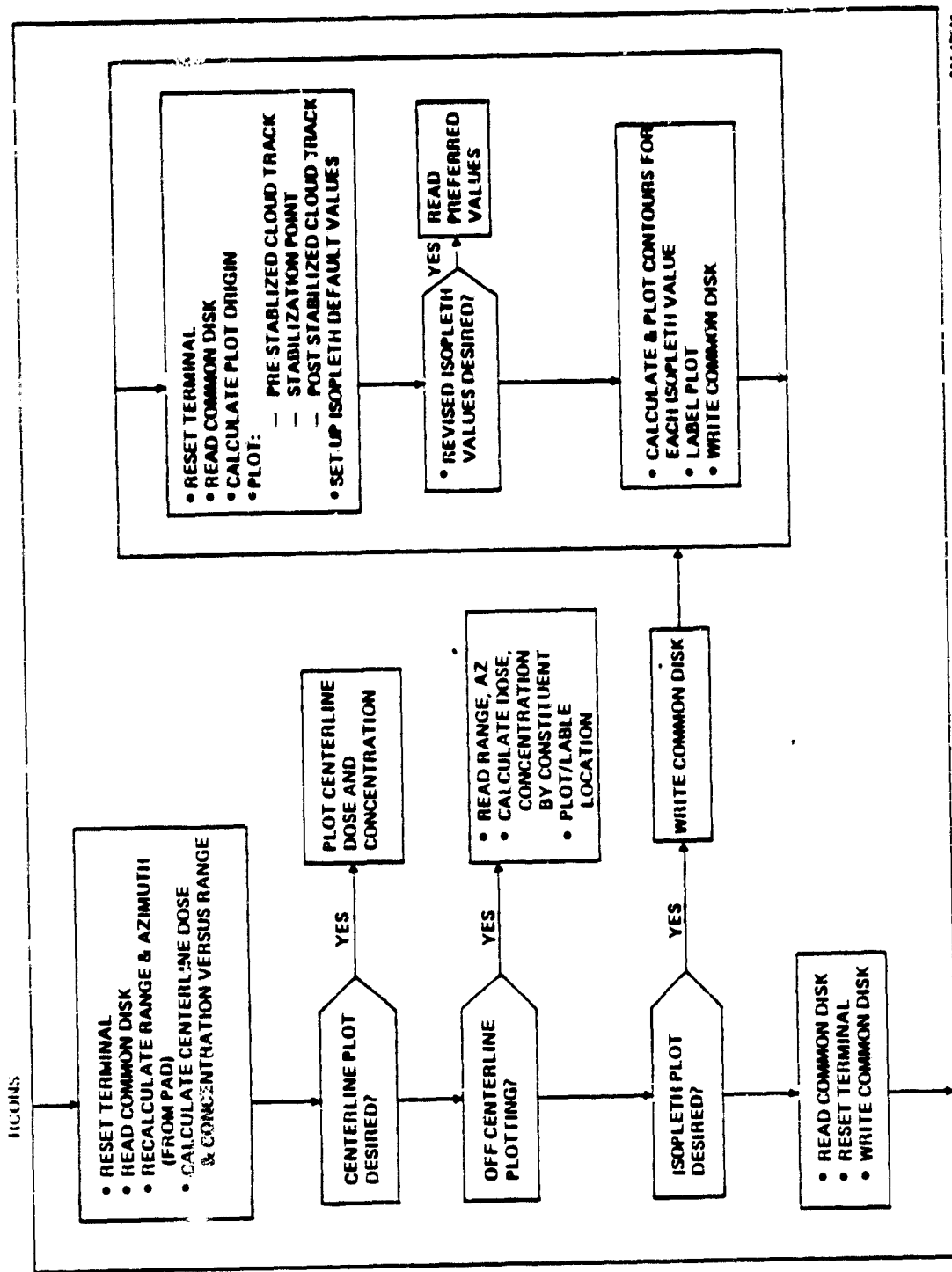




SAI-12815

FIGURE 2-3. (Continued) STRUCTURED DIAGRAM OF REEDS CODE





SAJ 12516

FIGURE 2.3. (Concluded) STRUCTURED DIAGRAM OF REEDS CORN





the commenting in the code as "landmarks", use will be made of the structured diagram of Figure 2-3 for guidance in module description. Each major calculational and structural event or user input option will be noted as a paragraph heading throughout these descriptions.

Common Data. The REEDS code makes use of bulk or "blank" common to simplify the use of subroutines. A penalty was paid for this in the form of a slight loss of modularity and some pain and agony due to vulnerability to "side effects" (indirectly and inadvertently changing a variable during a modification/upgrade). The penalty was lessened by structuring the common into testable "pseudo" blocks (not to be confused with block or labeled common), and by the use of descriptive comments for each significant "pseudo" block. This data is shared by the module and some of its subroutines, and is written to a "common" disk file. The common disk file is created by the program with the name "=REEDS" when the first attempt is made to write the common data. It is finally destroyed (purged) by the program upon successful termination of the program. If an abnormal termination leaves the file unpurged, it is, in principle, written over when it is detected to pre-exist; however, some strange results have occasionally been produced when the file pre-existed, prompting prudent users to be sure that it was purged before execution. Thus, following a brief statement of purpose, a date for last modification and any necessary loading instructions, a large block of common data (316 words) appears in each module. Just before another module is called this data is written on the disk file. Then, as soon as the new module is in core, the disk file is read, reloading the common. When the new module concludes, or prepares to call still another module, the common data is written again. A brief description of the common variables (Figure 2-4) appears in Table 2-1.

The REEDS "Question" Files. Several REEDS modules have their own question files, within which are stored informative messages and/or user prompts. These are indexed sequential coded (and therefore listable) files, program access to which is accomplished by calling subroutine "DREAD" (for Disk Read). The arguments to DREAD are:

- 1) A six (6) character, three-word file name

```

0036 C
0037 C >>> COMMON AREAS
0038 C
0039 C
0040 C >>> MATH PARAMETERS
0041 C COMMON PI,PIOVR2,PI43,TWOPI,SQR2PI,DEGRAD,RADDEG,IV2
0042 C
0043 C >>> VEHICLE PARAMETERS
0044 C COMMON VPAK(18),SIGHCL
0045 C
0046 C >>> OPTION PARAMETERS
0047 C COMMON YPOS,YES,ISNOFO,IWHICL,IRUN,ICALC,IPLACE,ITIMEZ,IGOTHM,
0048 C NUMRUN,LU,NUM,ONLY MFORM,LUPRNT,ITDU,IPL1,IPL2,IPL3
0049 C INTEGER YES
0050 C
0051 C >>> MET DATA
0052 C COMMON ALT(30),IDIR(30),SPEED(30),TEMP(30),PRESS(30),PTEMP(30),
0053 C SURDEN,FILE(3)
0054 C INTEGER FILE
0055 C
0056 C >>> CALCULATION TIME
0057 C COMMON ITIME,IDAY,INONTH(2),IYEAR
0058 C
0059 C >>> SOUNDING TIME
0060 C COMMON ISTE,ISDAY,ISMON(2),ISYEAR
0061 C
0062 C >>> LAUNCH TIME
0063 C COMMON LTIME,LTIM,LDAY,LMONTH(2),LYEAR,LAUNTD(10)
0064 C LOGICAL LTIME, LVERSH
0065 C
0066 C >>> CLOUD STABILIZATION AND LAYER PARAMETERS
0067 C COMMON STBALT,STBHT,STBAZD,STBRNG,STBTIM,CLDRAD,RISTIM(30),BOTLAY,
0068 C TUPLAY,CALHT,LAYTOP,LAYBOT,REFLEC,IPTZ
0069 C
0070 C >>> DIFFUSION COEFFICIENTS
0071 C COMMON SIGX0,SIGX,SIGZ0,SIGAP,XDIST,YDIST,EXPZ,SOSIGZ,SIGY,SIGAL,
0072 C CLDLNG,AVSPD,AVDIR,SIGAZ
0073 C
0074 C >>> CONCENTRATION AND USAGE VALUES AND RANGES
0075 C COMMON CONMAX,CONCPK,DOSMAX,DOSPK,RNGSHC,FIGSMD
0076 C

```

Figure 2-4. Program Excerpt Showing the Common Data Arranged in "Pseudo" Blocks



TABLE 2.1 COMMON DATA DESCRIPTION

"PSEUDO" COMMON BLOCK	PROGRAM LABEL/EQUIVALENCE NAME	SYMBOL	DESCRIPTION
MATH PARAMETERS	PI	$\pi$	3.141593
	PIOVR2	$\pi/2$	.5 * $\pi$
	PI43	$\frac{4}{3} \pi$	1.33333 * $\pi$
	TWOPI	$2\pi$	2.0 * $\pi$
	SQR2PI	$\sqrt{2\pi}$	$\sqrt{2\pi}$
	DEGRAD RADDEG IV2	$\pi/180$ $180/\pi$ —	DEGREES-TO-RADIANS CONVERSION PARAMETER RADIANS-TO-DEGREES CONVERSION PARAMETER CONVENIENT INDEX
VEHICLE PARAMETERS	VPAR(1)/QC1	$\dot{Q}_{C1}$	TOTAL SOURCE OUTPUT RATE (g/sec), NORMAL BURN
	VPAR(2)/QC2	$\dot{Q}_{C2}$	TOTAL SOURCE OUTPUT RATE (g/sec), ABNORMAL-SINGLE ENGINE BURN
	VPAR(3)/QC3	$\dot{Q}_{C3}$	TOTAL SOURCE OUTPUT RATE (g/sec), ABNORMAL-EXPLOSION/BURN
	VPAR(4)/QT1	$Q_{C1}$	TOTAL SOURCE STRENGTH (g), NORMAL
	VPAR(5)/QT2	$Q_{C2}$	TOTAL SOURCE STRENGTH (g), ABNORMAL-SINGLE ENGINE
	VPAR(6)/QT3	$Q_{C3}$	TOTAL SOURCE STRENGTH (g), ABNORMAL-EXPLOSION
	VPAR(7)/AA	a	*1st ROCKET RISE POWER LAW COEFFICIENT
	VPAR(8)/BB	b	*2nd ROCKET RISE POWER LAW COEFFICIENT
	VPAR(9)/CC	c	*3rd ROCKET RISE POWER LAW COEFFICIENT
	VPAR(10)/HEATN	$Q_{I1}$	EFFECTIVE HEAT RELEASE (cal/g), NORMAL BURN
	VPAR(11)/HEATM	$Q_{I2}$	EFFECTIVE HEAT RELEASE (cal/g), ABNORMAL-SINGLE ENGINE BURN
	VPAR(12)/HEATA	$Q_{I3}$	EFFECTIVE HEAT RELEASE (cal/g), ABNORMAL-EXPLOSION
	VPAR(13)/PHCL	f <sub>HCL</sub>	PARTIAL FRACTION OF HCL
	VPAR(14)/PCO	f <sub>CO</sub>	PARTIAL FRACTION OF CO
	VPAR(15)/PCO <sub>2</sub>	f <sub>CO<sub>2</sub></sub>	PARTIAL FRACTION OF CO <sub>2</sub>
	VPAR(16)/PAL2O3	f <sub>Al<sub>2</sub>O<sub>3</sub></sub>	PARTIAL FRACTION OF Al <sub>2</sub> O <sub>3</sub>
	VPAR(17)/PNO	f <sub>NO</sub>	PARTIAL FRACTION OF NO
	VPAR(18)/GAMMAX	$\gamma$	CLOUD RISE ENTRAINMENT COEFFICIENT
SIGHCL	$\sigma_{HCL}$	CONCENTRATION STANDARD DEVIATION OF HCL	

DAI-12618



TABLE 2-1. Continued

PSEUDO COMMON BLOCK	PROGRAM LEVEL EQUIVALENCE NAME	SYMBOL	DESCRIPTION
OPTION PARAMETERS	YPOS	---	CURRENT TERMINAL DISPLAY CURSOR ORDINATE
	YES	---	LOGICAL "YES" CODE (= 1 HY)
	ISNOFO	---	METEOROLOGICAL DATA TYPE INDICATOR 0 = SOUNDING 1 = FORECAST
	IVHICL	---	ROCKET VEHICLE INDEX  (1, 2, 3, 4, 5 FOR SPACE SHUTTLE, TITAN, DELTA, DELTA 3914, AND MINUTEMAN, RESPECTIVELY)
	IRUN	---	PROGRAM EXECUTION MODE (OPERATIONAL PRODUCTION)
	ICALC	---	CALCULATION FLAG
	IPLACE	---	METEOROLOGICAL DATA LOCATION CODE
	ITIMEZ	---	METEOROLOGICAL DATA TIME ZONE CODE
	IGOTMP	---	FLAG TO INDICATE WHETHER OR NOT A MAP ORIGIN HAS BEEN SET
	NUMRUN	---	TOTAL NUMBER OF RUNS (MET DATA FILE TO PROCESS)
	LU	---	INTERACTIVE TERMINAL DEVICE CODE
	NUM	---	RUN SEQUENCE NUMBER
	MONLY	---	METEOROLOGICAL PLOT SWITCH  1 = YES 0 = NO  USED IF ONLY PURPOSE OF RUN IS MET PLOTS
	MFORM	---	METEOROLOGICAL PLOT FORM SWITCH  1 = YES 0 = NO
	LUPRNT	---	PRINTING LOGICAL DEVICE KEY  6 = PRINTER OUTPUT, ELSE NO PRINTER OUTPUT
	ITDU	---	TOWER DATA SWITCH  0 = NO TOWER DATA 1 = TOWER DATA
	IPL1	---	METEOROLOGICAL PLOTTING DEVICE CODE
	IPL2	---	CENTERLINE PLOTTING DEVICE CODE
	IPL3	---	ISOPLETH PLOTTING DEVICE CODE

SAI-12797



TABLE 2-1. Continued

PSEUDO COMMON BLOCK	PROGRAM LEVEL EQUIVALENCE NAME	SYMBOL	DESCRIPTION
LAUNCH TIME	LTIME	—	LAUNCH TIME FLAG (TRUE IF LAUNCH TIME READ; OTHERWISE, FALSE)
	LTIM	—	LAUNCH TIME OF DAY (0001; 2400)
	LDAY	—	LAUNCH DAY OF MONTH (01; 31)
	LMONTH	—	2 WORD ARRAY FOR LAUNCH MONTH OF YEAR (e.g., 2HJA, 1HN)
	LYEAR	—	LAUNCH YEAR (e.g., 1981)
	LAUNTD	—	ARRAY OF CHARACTERS COMPRISING THE LAUNCH DATE AS READ FROM THE TERMINAL
CLOUD STABILIZATION AND LAYER PARAMETERS	STBALT	H	STABILIZATION HEIGHT (METERS)
	STBHT	$H_{SE}$	EFFECTIVE STABILIZATION HEIGHT (METERS)
	STBAZD	$\theta_s$	AZIMUTH FROM PAD OF STABILIZATION (DEGREES)
	STBRNG	$R_s$	RANGE FROM PAD OF STABILIZATION (METERS)
	STBTIM	$t_s$	TIME FROM LAUNCH OF STABILIZATION (SECONDS)
	CLDRAD	$R_c$	RADIUS OF CLOUD (METERS)
	RISTIM	$\Delta t_i$	30-WORD ARRAY OF TIMES FOR PASSAGE OF CLOUDS THROUGH EACH MET. LAYER
	BOTLAY	$Z_B$	HEIGHT OF BOTTOM OF LAYER CONTAINING STABILIZATION ALTITUDE
	TPLAY	$Z_T$	TOP OF LAYER CONTAINING STABILIZATION
	CALHT	Z	CALCULATION HEIGHT ( $Z_B \times Z \leq Z_T$ )
	LAYTOP	—	INDEX OF LAYER TOP ARRAY ELEMENT CONTAINING STABILIZATION
	LAYBOT	—	INDEX OF LAYER BOTTOM ARRAY ELEMENT CONTAINING STABILIZATION
	REFLEC	$\gamma$	SURFACE REFLECTION COEFFICIENT 0 = TOTAL REFLECTION 1 = TOTAL ABSORPTION
	IPTZ	—	LEAST SQUARE FIT FLAG FOR POTENTIAL TEMPERATURE GRADIENT CALCULATION 1 = NO FIT 0 = USE THE FIT

SAI-12760



TABLE 2-1. Continued

PSEUDO COMMON BLOCK	PROGRAM LEVEL EQUIVALENCE NAME	SYMBOL	DESCRIPTION
MET DATA	ALT	Z	ARRAY OF LAYER-TOP ALTITUDES IN THE MET PROFILE
	IDIR	$\theta$	ARRAY OF WIND DIRECTIONS OF THE LAYERS OF THE MET PROFILE
	SPEED	u	ARRAY OF WIND SPEEDS OF THE LAYERS OF THE MET PROFILE
	TEMP	T	ARRAY OF TEMPERATURES OF THE LAYERS OF THE MET PROFILE
	PRESS	P	ARRAY OF PRESSURES OF THE LAYERS OF THE MET PROFILE
	PTEMP	PT	ARRAY OF POTENTIAL TEMPERATURES OF THE LAYERS OF THE MET PROFILE
	SURDEN	$\alpha_0$	SURFACE DENSITY OF THE MET PROFILE
	FILE	--	FILE NAME ARRAY (8 CHARACTERS, 3 WORDS) OF THE MET FILE
CALCULATION TIME	ITIME	--	CALCULATION TIME (OF DAY (0001; 2400))
	IDAY	--	CALCULATION DAY OF MONTH (01:31)
	IMONTH	--	2-WORD ARRAY FOR CALCULATION MONTH OF YEAR (e.g., 1981)
SOUNDING TIME	ISTIME	--	SOUNDING TIME OF DAY
	ISDAY	--	SOUNDING DAY OF MONTH (01:31)
	ISMONTH	--	2-WORD ARRAY FOR SOUNDING MONTH OF YEAR (e.g., 2HFE, 1 HB)
	ISYEAR	--	SOUNDING YEAR (e.g., 1981)

SAI-12789



TABLE 2-1. Continued

PSEUDO COMMON BLOCK	PROGRAM LEVEL EQUIVALENCE NAME	SYMBOL	DESCRIPTION
DIFFUSION COEFFICIENTS	SIGXO	$\sigma_{X_0}$	STABILIZED CLOUD ALONG WIND STANDARD DEVIATION
	SIGX	$\sigma_X$	ALONG-AXIS 1-CLOUD EXTENT
	SIGAP	$\sigma_A'$	WIND AZIMUTH STANDARD DEVIATION AT STABILIZATION
	XDIST	X	ALONG-AXIS DISTANCE
	YDIST	Y	OFF-AXIS DISTANCE
	EXPZ	$\Sigma Z$	VERTICAL DOSAGE TERM
	SOSIGZ	$2\sigma Z^2$	TWICE THE SQUARE OF THE 1- VERTICAL CLOUD EXTENT
	SIGY	$\sigma_Y$	OFF-AXIS 1-CLOUD EXTENT
	SIGAL	$\sigma_{AL}$	LAYERED-AVERAGED WIND AZIMUTH STANDARD DEVIATION
	CLDLNG	$C_L$	CLOUD LENGTH (METERS)
	AVSPD	U	SPEED OF CLOUD AVERAGED OVER MET. LAYERS (METERS/ SEC)
	AVDIR	$\theta$	DIRECTION OF CLOUD AVERAGED OVER MET LAYERS (DEGREES)
	SIGAZ	$\sigma_{AZ}$	WIND AZIMUTH UNCERTAINTY (DEGREES)
	CONCENTRATION AND DOSAGE VALUES AND RANGES	CONMAX	$\chi_M$
CONCPK		$\chi_p$	PEAK CONCENTRATION (PPM)
DOSMAX		$D_M$	MAXIMUM PEAK DOSAGE (PPM-SEC)
DUSPK		$D_p$	PEAK DOSAGE (PPM-SEC)
RNGSMC		$R_{Mc}$	RANGE OF MAXIMUM CONCENTRATION (FROM PAD)
RNGSMD		$R_{MD}$	RANGE OF MAXIMUM DOSAGE (FROM PAD)

SAI-12700

- 2) The desired line number (index + 1)
- 3) A sixty-four (64) character (32 word) data buffer to receive the ASCII data.

The first line of each question file is an eighty (80) character sequence of the numbers

1234567890123456789012. . .

useful for preparation/editing of the question file itself (i.e., aids in columniation of characters), so the actual line number is always one (1) greater than the index of useful data to be retrieved from the question file. Section 4.3 shows the REEDS question file listing. Special CRT display enhancement characters are often embedded as part of the character strings. They are unprintable with the selected print option used in generating the listings of Section 4, and do not show up even as blank spaces. This gives an erroneous column indication for characters which do show up. However, the special Character Display feature of all HP CRT's permits display of the enhancement character symbols, facilitating editing of the lines of these files. Once read by DREAD, these lines are printed on the desired device by subroutine "CHAR", which performs the necessary terminal-dependent operations required to display the line read.

Communication of Data To (Or From) Output Files. When it is desirable to communicate program numeric data to (encoding) an output file, or from (decoding) that file, it must be first internally converted to (or from) an ASCII character string. Encoding is accomplished with a call to the system subroutine "CODE" and a subsequent specialized "READ" operation. Decoding requires the use of "WRITE", instead of "READ". The optional argument to CODE is the number of characters desired to be encoded (or decoded). Many painful agonies were avoided by explicitly defining and using this argument. Also great debugging horrors were generated by inadvertently putting some other statement between the call to "CODE" and its subsequent "READ" (or "WRITE"), or otherwise failing to resolve a call to "CODE" with the necessary "READ" (or "WRITE") call. (These are mentioned here in the event users use this document as a reference in making any changes to REEDS.) The arguments to "READ" (or "WRITE") are normally the standard output device variable and the FORMAT code. However, when "READ"





(or "WRITE") is used to resolve a call to "CODE" to encode (non-ASCII) or decode (ASCII) data, the first argument is the array variable name used to receive the encoded (or send the decoded) data. The size of this array must be compatible with the FORMAT statement (the 2nd argument) used to accomplish the "READ" (or "WRITE"). Once encoded, the ASCII string is written to an output, whenever it is desired, in the normal fashion, i.e., with an "A" FORMAT formatted "WRITE" call. If the data were decoded (via the "WRITE" call) instead, the resulting data (FORMAT permitting) could be arithmetically or otherwise processed, and must be re-encoded if the ASCII string is needed.

### 2.1.1 The REEDS Executive Module

The REEDS executive module

- Initializes some global operational, terminal control, and calculational data
- Reads additional control and parameteric data
- Reads and prepares the meteorological data
- Calculates the potential temperature profile
- Controls subsequent scheduling of modules.

The bulk of the module consists of interactive option selection and related parameter initialization. The main program (file "REEDS") is 1157 source lines long at this printing, and the subroutine packages required (files "REEDT", "LINE", "DLINE", "CALIX" and "CTLB2") constitute another 1555 source code lines. Refer to Figure 2-3 as necessary to follow the program structure.

User Terminal Identification Setup. The type of terminal (CRT, Plasmascop) is first discerned by the program from system tables internal to the HP1000 operating system. Some terminal parameters are subsequently initialized -- automatically and transparent to the user. The header is transferred from the REEDS executive module question file (called "=REEDS") to the terminal display.

Cloud Rise Fit or Virtual Potential Temperature Lapse Rate Fit Option. The desired fit to the Lapse Rate of Virtual Potential Temperature is then produced for the cloud rise model as a result of a user option. The choice is between a deterministic method or a least squares method suggested by

EETT leader Dr. J. B. Stephens. The latter method produces consistently lower altitudes than the former for cloud rise stabilization and has thus been termed the "low altitude fit". This choice remains as a user option pending documented validation of one method or the other. This option was installed after Plasmascope operation was largely phased out. Thus, the normal terminal independent input sequence is not used, rather, a direct CRT terminal input is used.

Number of Runs and Input Meteorological Data File Names. The REEDS program was designed to process multiple meteorological data files if that is desired. The procedure is to set up a series of data files which have in common the first four (4) characters of their names, and a relative sequence number, i.e., 01, 02, . . . 99, for their last 2 characters. When prompted to do so, the user enters:

XX, FFFFYY

for the number of runs, data file name root, and code YY, respectively, where

XX = Number of runs desired [01 ; 99]  
FFFF = The first four (4) characters of the series  
of meteorological data file names  
YY = The file of the series it is desired to start  
with, e.g., DATA01

The program has a default data file name root of "DATA" which may be invoked by either assenting to the use of the default (which also assumes there will be only 1 run of DATA01), or by entering blanks for FFFF ( which causes any entry for YY to be ignored, presetting its value to 1, thus causing the series DATA01 DATAXX to be executed). One run of an arbitrary file name is accomplished by declining the default, and entering

XX, FFFFYY



where XX = 01, and FFFFYY is the six (6) character file name desired, e.g.,

01, TESTFZ

Geographical Location and Time Zone Codes. As a result of a standard NASA naming convention the geographical location, and therefore the time zone code, can be embedded within the meteorological data file name. These are the common variable ITIMEZ and IPLACE, respectively. The following illustrates the implications of certain non-standard values for FFFF:

FFFF = VAND (Vandenberg AFB), IPLACE = 1, ITIMEZ = 1HP  
FFFF = KSC (1965 data tape), IPLACE = 2, ITIMEZ = 1HE  
FFFF = KSC, IPLACE = 3, ITIMEZ = 1HE  
FFFF = SPEC, Special location, IPLACE and ITIMEZ = 0, and actual subsequently input according to the format: values  
:XX,YYZ,AAAA

where

XX = Day of month [01 ; 31]  
YYY = Month [Jan ; Dec], and  
AAAA = Time [0001 ; 2400].

Any other input is assumed to be the standard KSC meteorological data file with ITIMEZ = 1HE, and IPLACE = 2. The meteorological data file can only be an actual sounding for FFFF = SPEC, whereas for other inputs sounding and/or TOWER data may be used.

Research, Operational, or Production Modes. As of the time this document was being generated the only validated mode of these alternatives was the operational mode -- which is also the default and most commonly used mode. While the additional user inputs inherent to the research mode will be mentioned, their demonstration in Section 3 will have to await a subsequent revision of this document. Nevertheless, the appropriate non-default selection is made by entering a single character R, O, or P, respectively. This results in the

setting of common variable IRUN to 1, 2, or 3, respectively. Default selection is, of course IRUN = 2.

Atmospheric Mixing Layer Top and Wind Azimuth Uncertainty - For Production Runs. For production runs, the input of the top of the atmospheric mixing layer, TOPLAY, (in meters) and the standard deviation of the wind azimuth uncertainty, SIGAZ, (in degrees) must be input from the keyboard. The format, upon prompt is

XXXX.XX

for TOPLAY, with possible range [0;999999.0] meters, followed by

XX

for SIGAZ, with possible range [0;99] degrees. For the other operational modes (Research, Operational) the values of TOPLAY and SIGAZ are estimated and then optionally changed by the user. This will be elaborated upon when it is encountered in the RCLDS cloud rise module (Section 2.1.2)

Meteorological Forms Switch. The REEDs program plots some of the meteorological data (called a "Met Plot") and a schematic of the stabilized exhaust cloud. The form for this plot can also be prepared by the program on the plotter, but the process is very time consuming. Therefore, it is better done as infrequently as is necessary and high-speed reproduction devices used to generate a supply of the forms. For the purpose of generating this form, the REEDS program may be run to the point of this input for the sole purpose of generating the form. The run may then be terminated and subsequently restarted once a supply of forms is available. Thus, when prompted for "Met Plot Forms?", the user may respond "Y" for 'Yes' or "N" for 'No', the latter of which is the default response. This sets the common (OPTION pseudo block) parameter MFORM to 1 or 0, respectively.

Meteorological Plots. The meteorological plot may also be skipped if it is desirable. When prompted, a "Y" or "N" indicates 'Yes' or 'No', respectively. This sets the common (OPTION pseudo-block) parameter MONLY to 1 or 0, respectively. The first setting is the default setting.



Line Printer Output. The line printer output may be omitted if desired. When prompted a "Y" or "N" indicates 'Yes' or 'No', respectively. This sets the common (OPTION pseudo-block) parameter to 6 or  $64_{10}$  respectively. The first setting is the default setting, and its value of 6 indicates the standard printing device code. The value of  $64_{10}$  for the alternate setting formerly cleared the parameter (largest value was  $77_8 = 63_{10}$ ). Now, however, the device code range has been extended above  $100_8$ , making  $64_{10}$  a legal device code. However, no such code is active on the MSFC or KSC machines, so no change (e.g.,  $64_{10} \rightarrow 128_{10}$ ) has yet been made.

Launch Time and Date. The current computer system time and date are loaded into the launch time and date variables, LTIM, SDT, LDAY, LMONTH, LYEAR via the call to the subroutine GETDT, which reads the system clock. The user is prompted for this information, and if the response to the prompt is "N" for 'No', the system time/date is rejected as the launch time, and opportunity is given for the above variables to be read accordingly to

TTTTbbbbDDbMMbYYYY

where TTTT = the time of day [0001;2400] ,  
b or bbbbb = blank space place holders indicating the number of user-  
input blank spaces  
DD = the day of the month [01;31]  
MM = the month [JAN;DEC]  
YYYY = the year (e.g., 1980)

If the response to the original prompt is "Y" for Yes or blank, the system time is accepted for the default launch time/date, and the time/date variables are left as initialized.

Launch Vehicle. The user is then prompted for the launch vehicle desired. The options are "D" for Delta, "T" for Titan, and "S" for Shuttle. This sets the common (OPTION pseudo-block) variable IVHICL to 0, 1, or 2, respectively.

Meteorological Data Input. If the IPLACE parameter, set by the meteorological data file name read above, is 2, then the forecast/sounding data



is assumed to reside on tape and is read from device code 8 via subroutine KSC65 (explained below). Otherwise the appropriate disk file name is opened. Whether the data represents an actual sounding or a forecast is indicated by a code read from the file header. If a sounding is indicated, the data type parameter, ISNDF0, (an OPTION pseudo-block common variable) is set to 0; if it is a forecast, it is set to 1. As the header and/or data lines are read, they are printed on the printer device. The following indicates the sequence of data lines read.

- (1) test code, data type and location header information
- (2) time of data, ascent and data header information
- (3) actual data for each altitude:
  - altitude (feet)
  - wind direction (degrees from north)
  - wind speed (knots),
  - temperature ( $^{\circ}\text{C}$ ),
  - Dew Point ( $^{\circ}\text{C}$ ),
  - pressure (millibars)
  - relative humidity (percent)
  - Absolute humidity ( $\text{gms}/\text{M}^3$ ) (if present on file)
  - Air density ( $\text{gms}/\text{M}^3$ ) (if present on file)
  - other data (if present) (ignored by program)
- (4) Terminator (NNNN) indicating end of data. (See Section 4.4)

The program is designed to ignore all non-essential data words and/or data lines. The altitude and wind velocity are then converted to other units; they are converted as follows:

altitude — m  
wind speed — m/sec

All data lines with altitudes less than 10000 feet are then sorted by ascending order of altitude. The potential temperature is then calculated according to

$$PTEMP_i = (TEMP_i + 273.6) \left( \frac{1000}{PRESS_i} \right)^{.288}$$

where PTEMP, TEMP, and PRESS are the COMMON variables (MET data pseudo-block) potential temperature, temperature, and pressure, respectively. The sorted data is then written on the output device and the meteorological data file closed.

Met File Reads Via Subroutine KSC65. The subroutine KSC65 handles MET file reads from tape (device code 8). The sequence of data is identical to that of the disk file read. The subroutine was designed to read large data bases, especially the 1965 KSC meteorological data tapes (thus the subroutine name). The arguments to KSC65 facilitate selection of specific data sets from the tape. They are as follows:

- IBUF = Eighty (80) character (40 word) integer buffer array to hold ASCII data lines read from device 8 pending decoding.
- IALT = Altitude data array to be passed back to the main program for eventual conversion to the analogous common array.
- DPTEMP = dew point temperature array to be passed back to the main program
- IFOFF1 = (KSC65 variable IWANT) = ordinal data set indicator for selection among data sets of the same date/time.
- IEOF = End of File (EOF) flag set if EOF encountered on device 8 (set by Equipment Table Check Three system EXEC call)
- ITIMG = Sounding time desired
- IDAYG = Sounding day desired
- IMONG = Sounding month desired
- ISNDG = data type desired (sounding/forecast)
- RUNNUM = run numbers desired

Creation of the Common Data Disk File. The common data, though not completely defined yet, is written to the disk file "=REEDS" via a call to the subroutine "RWDIS." The two arguments to RWDIS are the six (6) character common data disk file name (3 integer words, 2 characters each), and the operation integer code (1 = write).



RWDIS contains the 316 word common block as it exists in the main program. In addition, space is provided for the data control block array (locally called ODCB) necessary to control the disk read (transparent to user - only provision of the space necessary, i.e., 144 integer words or more for the control array). The data is transferred to/from disk via a buffer array. While the common data transferred is 316 words, disk storage is only by integer word equivalents. Thus, because of the real data variables/arrays, the integer word equivalent of the 316 words is 557 words. The buffer array is thus 557 words.

The ISIZE array, which is the parameter controlling the file creation operation, must have its two integer words set to 30 and 557, respectively. The first word is the number of blocks the 557 words are written in.

The file is first opened. If it is not detected to pre-exist, the file is created with the CREAT system call. For a RWDIS operation code of 1, the file is written; if 0, the file is read. The file is then closed.

Transfer to the Cloud Rise Module. As can be seen in Figure 2-3, transfer to the Cloud Rise Module RCLDS then occurs. The transfer is accomplished via the system EXEC call, which schedules the program whose name is contained in a six (6) character (3 word) argument array (RCLDR). The program must have an ID segment in CORE, an operation accomplished via the File Manager RP command which must be executed before the running sequence begins (see File Manager or Terminal Users manual).

Program Termination. At the conclusion of RCLDS, transfer is returned to REEDS and the user is prompted to decide for program termination ("Y" or default response) or for the processing of more data ("N" for No termination). If the latter response is chosen, REEDS begins again. Otherwise, the CRT screen is cleared, and the common disk file is purged. Finally, all program ID segments in the REEDS series are removed from core ("OFF'ed" -- the reverse of the "RP" operation -- see File Manager or Terminal User's manual). The program STOP is executed and REEDS is terminated.

### 2.1.2 The RCLDS Cloud Rise Module

The RCLDS Cloud Rise Module models the thermodynamic phase of the REEDS code. In so doing, the meteorology, energy, and potential temperature factors





are calculated. Then the range from the pad of the cloud center, the cloud transport direction, stabilization height, and surface mixing layer depth are calculated. These are necessary for the formation of one of the more significant calculations -- the wind azimuth uncertainty. This parameter is an important driver of the turbulent diffusion process. Once the surface mixing layer height is estimated and optimally refined, the RMTPS module is called to plot the pertinent geometric relationships and display the input meteorology and calculated potential temperature. The program is then ready for the kinetic phase of the calculation.

Data File Reads. The prompt for the surface layer height is transferred from the RCLDS question file, "?RCLDS" (Figure 4), to a holding array via the DREAD subroutine. The common disk file "=REEDS," is then read, some local variables and display flags initialized, and the exhaust cloud print header is written to the printing device.

Useful Constant Terms for the Meteorological/Energy Factors. In preparation for the calculation of the stabilized height and cloud rise energy factor, two terms useful for the calculation at a meteorological factor for bouyant rise,

$$GT = \frac{9.8}{T + 273.16}$$

and

$$\frac{.05129 (T - 273.16) P_0 \gamma}{Qc_1}$$

are calculated.

Adiabatic or Stable Cloud Rise. Next, a loop is performed through the meteorological layers to simulate the cloud rise phenomenon, either adiabatic or stable. The criteria is the argument of the inverse cosine (ARCCOS) term in the bouyant cloud use equation. If the argument is less than (-1), then the rise is "stable," and the flag so indicating, IAS, is set to 1; otherwise, the rise is "adiabatic," and IAS is 0. IAS is initialized in each layer to 1 ("stable" cloud rise).



Cloud Stabilization and Criteria for Surface Mixing Layer Height Selection. The slope (derivative with respect to altitude) of potential temperature (i.e., the "Lapse Rate"), the wind speed, and wind direction, in each layer are then calculated. The reader may recall the optional methods of calculating the lapse rate set up by the executive module, REEDS. If the least squares method (resulting in a lower cloud rise) is selected, a parameter IPTZ was set to 0. Otherwise, the alternate method of determining the lapse rate was chosen (IPTZ = 1), and is now calculated. In either case, a lapse rate is determined, and the energy factor calculated:

$$\alpha = \frac{\alpha_c(Z_i - Z_0)}{\Delta t_R},$$

where  $Z_i$  and  $Z_0$  are the current and initial layer altitudes, and  $\Delta t_R$  is the deposition time for the launch vehicle as defined by the vehicle's time versus altitude profile. The stability factor, STAB, is calculated from either lapse rate, determining adiabatic ( $STAB \leq 0$ ) or stable ( $STAB > 0$ ,  $[1 - \frac{1}{2} \alpha \cdot STAB] < -1$ ) rise. If the rise is adiabatic, the rise time through the current layer is merely

$$\sqrt{x};$$

otherwise it is 
$$\left[ \frac{\pi/2 - \tan^{-1}(C_3)}{\sqrt{STAB}} \right]$$

where  $C_3 = \frac{[1 - 1/2 \alpha \cdot STAB]}{\sqrt{1 - [1 - 1/2 \alpha \cdot STAB]^2}}$

In addition to determining cloud rise, the three slopes calculated above become input factors to a deterministic calculation of the estimated surface mixing layer height to be performed after the conclusion of the rise loop and final stabilization height calculation:

Stabilization Range and Transport Direction. As the cloud rises through the meteorological layers, the wind speed and direction change. This affects the cloud direction and resultant position, so these effects are determined. This results in a transport direction and a stabilization range



increment. At the conclusion of this loop, a stabilization range and height are calculated. Later, the range and direction will be averaged over the rise to produce a net direction throughout the single surface mixing layer.

Mixing Layer Height. Subroutine FTOP is then called to estimate the height of the surface mixing layer. To do this, the slopes of potential temperature, wind speed, and wind direction versus time are examined for "significant" changes. Just what is significant is somewhat subjective. Currently, three slope change magnitudes are stored in a data statement. They are:

$$\left| \Delta \frac{d(PT)}{dz} \right| = 0.03$$

$$\left| \Delta \frac{du}{dz} \right| = 0.015$$

$$\left| \Delta \frac{d\theta}{dz} \right| = 0.015$$

for potential temperature, wind speed, and wind direction, respectively. When these are exceeded by the respective slope changes, the changes are considered "significant."

If no significant changes are found, the mixing layer height is set to twice the stabilization height. If only one of the variables has a significant change, the altitude of that change is selected. If there are more than one, they are sorted in descending order of magnitude. If there is an altitude at which all three variables have significant changes, this altitude is chosen. If, instead, there are two with coincident altitudes of significance, that altitude is chosen. If there are multiple multi-variable altitudes of significance, or several different altitudes which are non-coincident, then the largest change is selected. The following list summarizes the selection priority:

- 1) The largest triple variable coincident significant change.
- 2) The largest double variable coincident significant change.

- 3) The largest potential temperature significant change.
- 4) The altitude of the single significant change.
- 5) Twice the stabilization altitude.

Wind Azimuth Uncertainty. A very important driver of the turbulent diffusion process is the wind azimuth uncertainty ( $\sigma_{Az}$ ). This variable is computed by the subroutine "RSGAZ," using a theoretical and deterministic method developed by Panofsky from similarity theory. The alternative to a method such as this would be a statistical method. However, with soundings often used as input meteorological sources, resolution at near-surface altitudes may not be good. This is greatly improved with tower data. Nevertheless, the similarity theory-based deterministic method is an important contribution for predictive application to single layer turbulent diffusion.

The value of  $\sigma_{Az}$  depends on the value selected from FTOP for the top of the surface mixing layer. That height is not final (i.e., it may be changed by the user if the Research or Operational mode is used, so final calculation of  $\sigma_{Az}$  occurs when the mixing layer height is accepted. The form used for  $\sigma_{Az}$  is

$$\sigma_{Az} = \frac{\sigma_v}{\bar{u}} = \frac{kf(B)}{\ln(z/z_0) - \psi(R_I)}$$

- $\sigma_v$  = standard deviation of lateral component of turbulence (m/sec)
- $\bar{u}$  = mean wind speed (m/sec)
- $k$  = Von Karman's constant (= .4)
- $f(B)$  = Piecewise linear fit to experimentally measured values of  $\sigma_v/\bar{u}$  as a function of the non-dimensional stability ratio,  $B$
- $z$  = altitude (m)
- $z_0$  = roughness length (m)
- $\psi(RI)$  = a universal function of the Richardson number.



This relation was derived by Lettau (unpublished) and Panofsky (1963), serving both stable and adiabatic atmospheres (International Journal of Air and Water Pollution, Vol. 9, 1965, Panofsky and Prusad on "Similarity Theory and Diffusion"). The non-dimensional stability ratio, B, is given by

$$B = \frac{g \bar{z}^2 \Delta(P_T)}{T \bar{U}^2 \Delta z}$$

where

- g = gravitational acceleration (m/sec<sup>2</sup>)
- $\bar{z}$  = the geometric mean of layer top and bottom (m)
- $\bar{U}$  = mean wind speed (this layer) (m/sec)
- T = absolute temperature (°K)
- $\Delta P / \Delta z$  = lapse rate (vertical gradient of potential temperature).

Depending on the value of B, f(B) is selected from

f(B) = a + b (B + c), where

$$\left\{ \begin{array}{ll} A = 2.7, b = c = 0 & \text{for } B < -.008 \\ = 2.7, b = 112, c = .008 & -.008 \leq B < -.00175 \\ = 3.4, b = 725.5, c = .00175 & -.00175 \leq B < .0008 \\ = 1.55, b = 38.04, c = -.0008 & .0008 \leq B < .029 \\ = 2.35, b = 5.43, c = .029 & .029 \leq B \end{array} \right.$$

The Richardson number for  $\psi(R_I)$  cannot be calculated from its definition (normalized lapse rate per unit vertical wind speed gradient squared), because sufficiently accurate measurements of the vertical wind speed gradient do not normally exist. It can be transcendently obtained from:

$$R_I = B \left[ \frac{\ln(z/z_0) - \psi(R_I)}{\phi(R_I)} \right]^2$$

where (RI) =  $(1-16 R_I)^{-1/4}$ , unstable conditions  
 $(1-7RI)^{-1}$ , stable conditions

(another universal function),



and

$$\begin{aligned}\psi(RI) &= \ln \frac{1+X}{2} + \ln \frac{1+X^2}{2} + \frac{\pi}{2} - 2 \tan^{-1}(x), \text{ unstable condition} \\ &= 7 R_I(1-7R_I)^{-1}, \text{ stable conditions}\end{aligned}$$

where  $X = (1-16RI)^{1/4}$ .

For stable conditions,

$$\sqrt{RI} = (-1 + \sqrt{1 + 4 A_1 A_3}) / 2A_1$$

where  $A_1 = 7 Bk$ , and  $A_3 = -B(k-1)$ , for  $k = \ln Z/Z_0$ .

For unstable conditions,

$$\frac{(1-x)^4}{16 X^2 [\ln z/z_0 + .50864 - 2 \ln(1+x) - \ln(1+x^2) + 2 \tan^{-1}x]} - B = 0$$

is solved transcendently for  $X$ , which yields  $RI$ .

In RSGAZ, the transcendental function statement and its necessary derivative precede the active statements (follow the detailed flow chart in Figure 2-5). The critical data (transferred via common) is selected from the input layers as a result of the input arguments passed. The potential temperature gradient (Lapse Rate) is then calculated at the indicated altitudes, followed by the Richardson number for either stable or unstable conditions. The appropriate value of  $f(B)$  is then obtained for the final calculation of  $\sigma_{Az}$ . The value of  $\sigma_{Az}$  returned is constrained between the limits of 7 degrees and 30 degrees, to keep resultant uncertainties physically realizable.

Interactive Alteration of the Surface Mixing Layer Height and the Wind Azimuth Uncertainty. Once the mixing layer height and subsequent wind azimuth uncertainty are estimated by the program, the user is given the opportunity to change each, if the RESEARCH or OPERATIONAL mode has been chosen. In the production mode, the user must enter both. The user is first prompted to use the estimated layer height or an altered value. If the altered value choice is



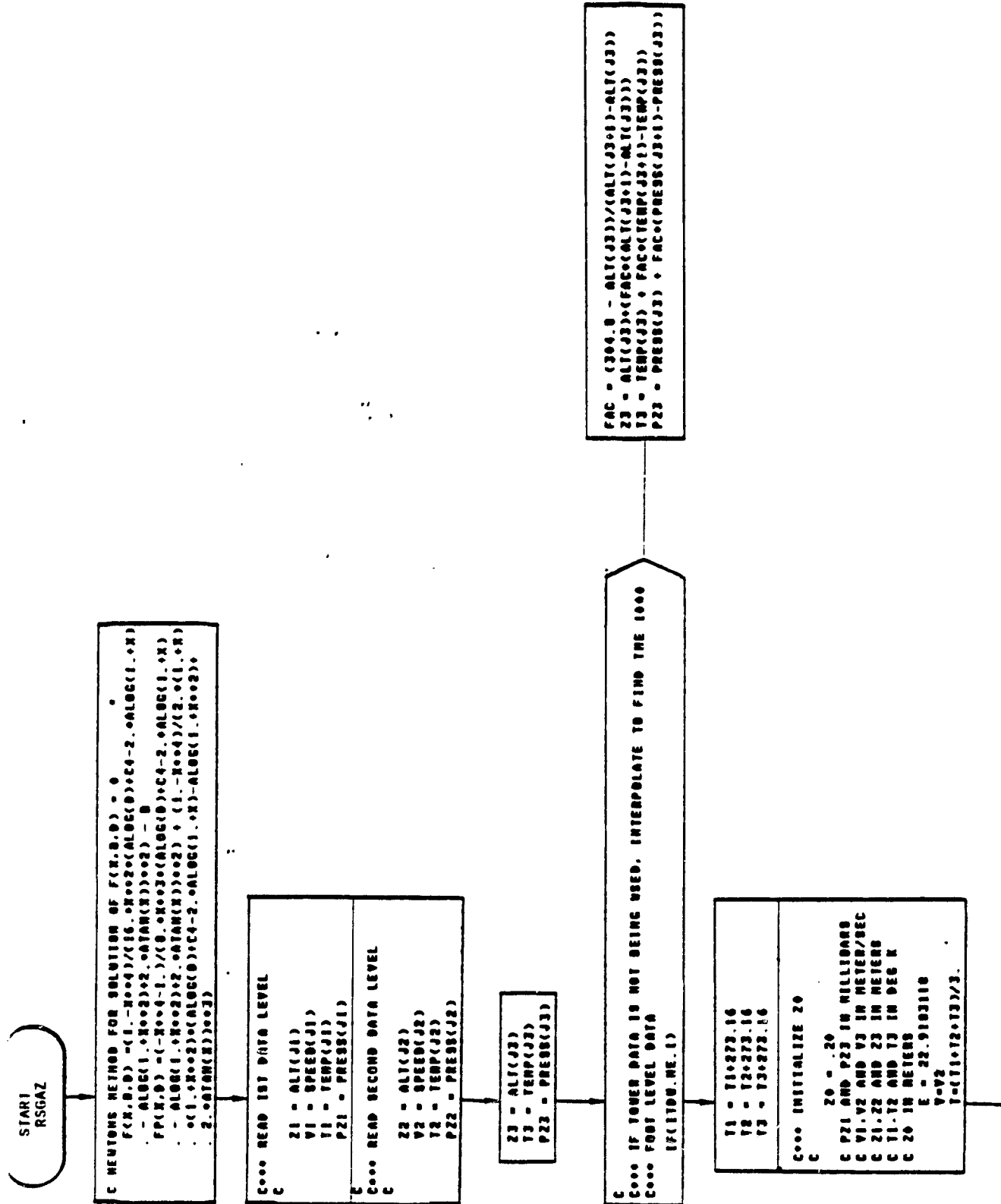


Figure 2-5. Detailed Structure Flow Diagram of Subroutine RSGAZ



ORIGINAL PAGE IS  
OF POOR QUALITY

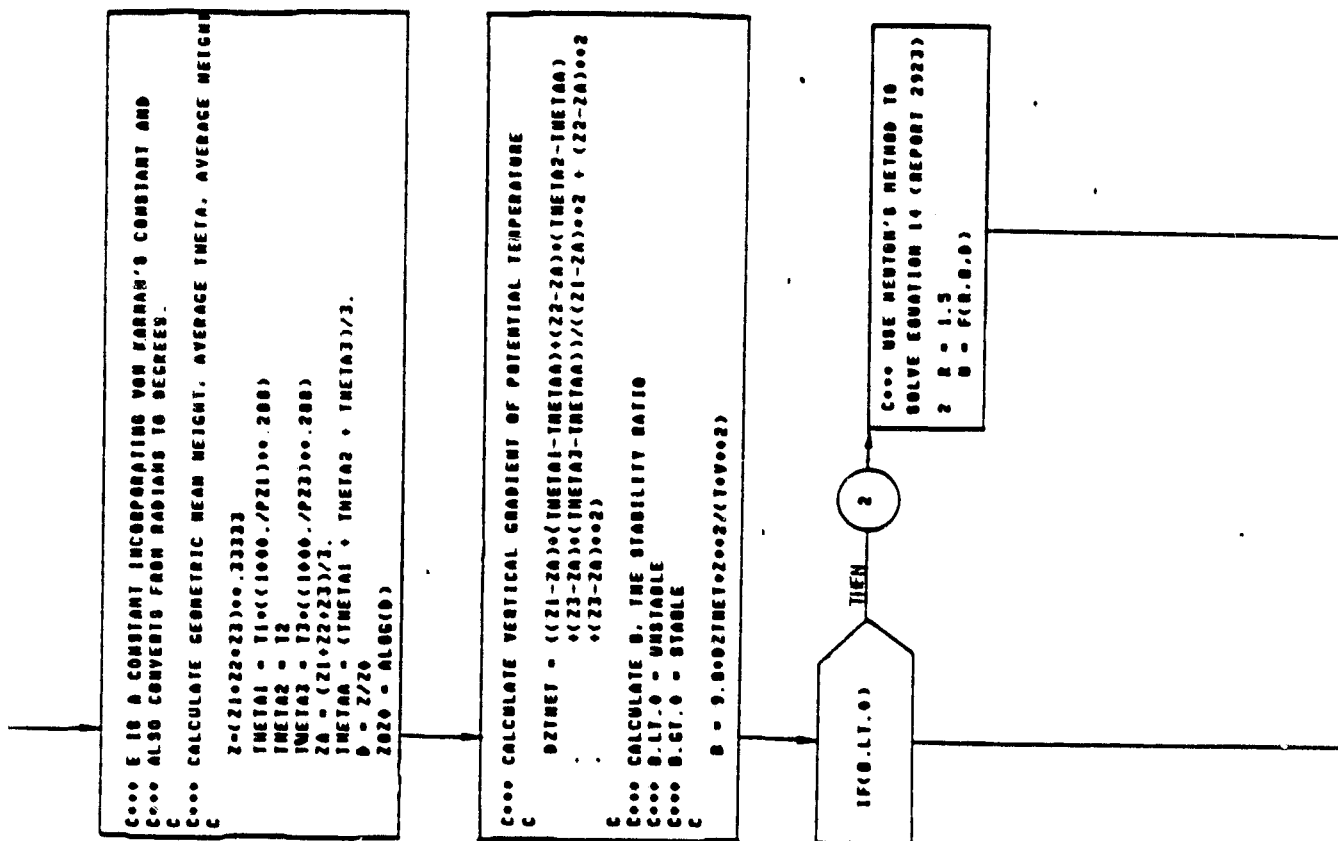


Figure 2-5. Detailed Structure Flow Diagram of Subroutine RSGAZ (continued)





ORIGINAL PAGE IS  
OF POOR QUALITY

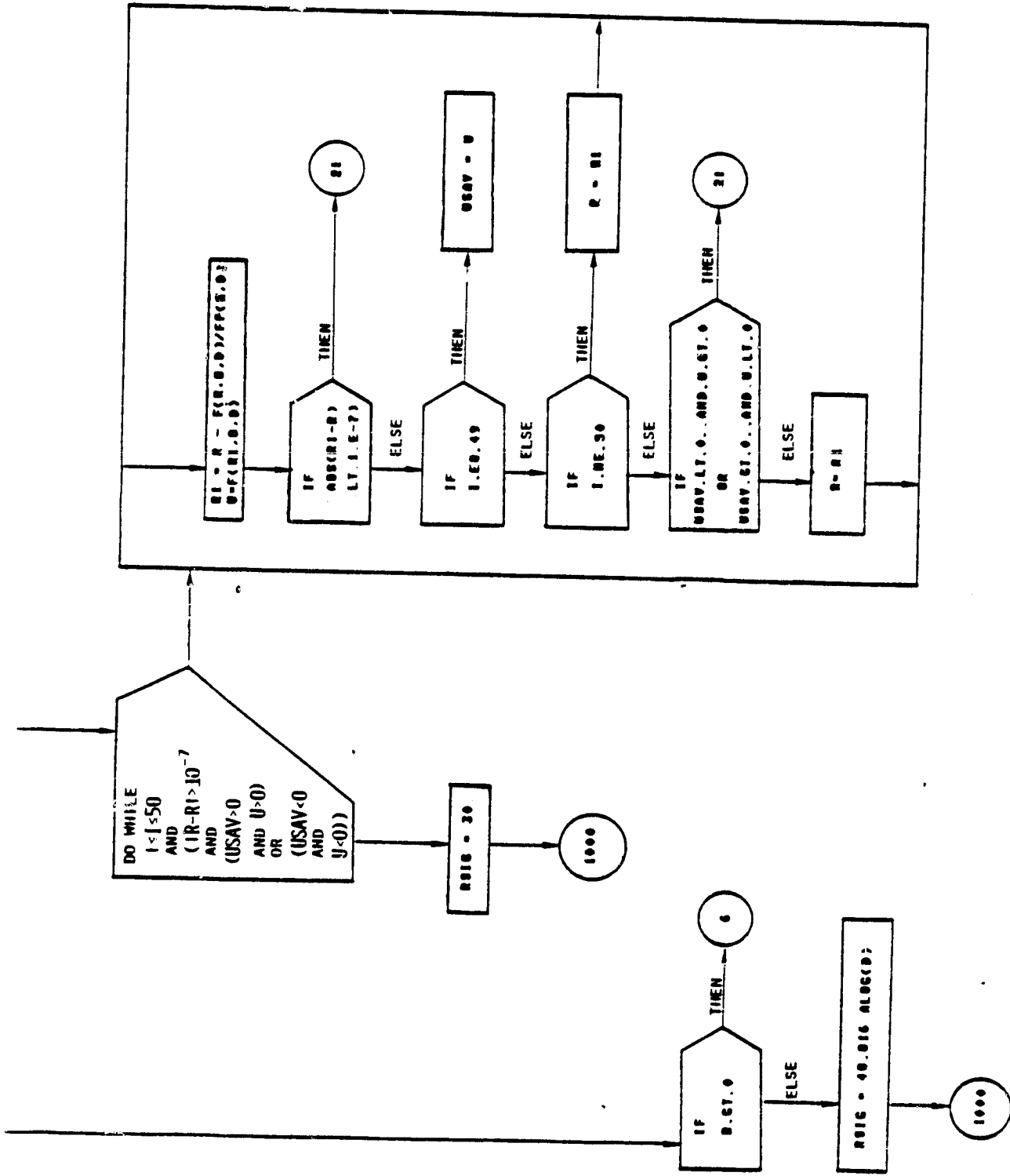


Figure 2-5. Detailed Structure Flow Diagram of Subroutine RSGAZ (continued)



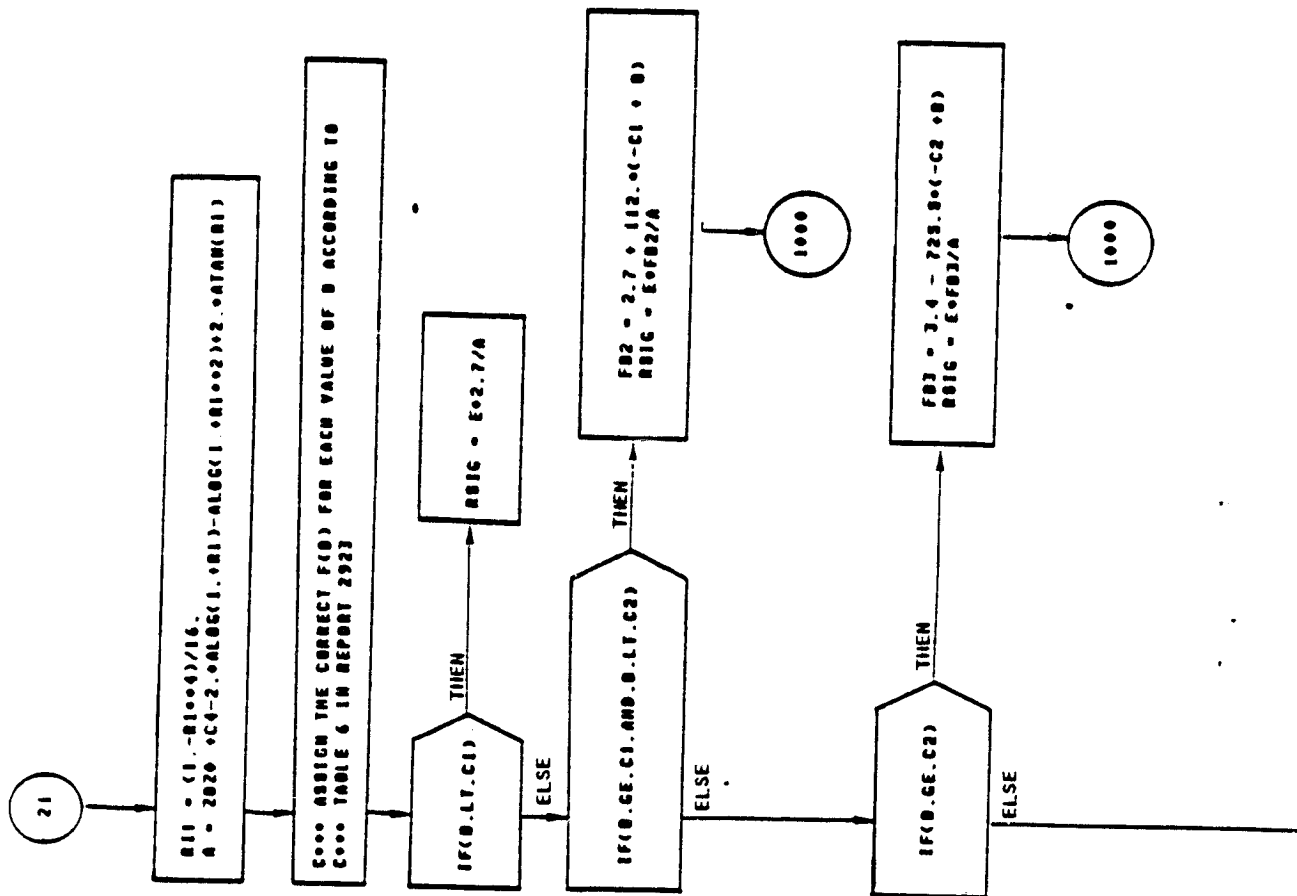


Figure 2-5. Detailed Structure Flow Diagram of Subroutine RSGAZ (continued)



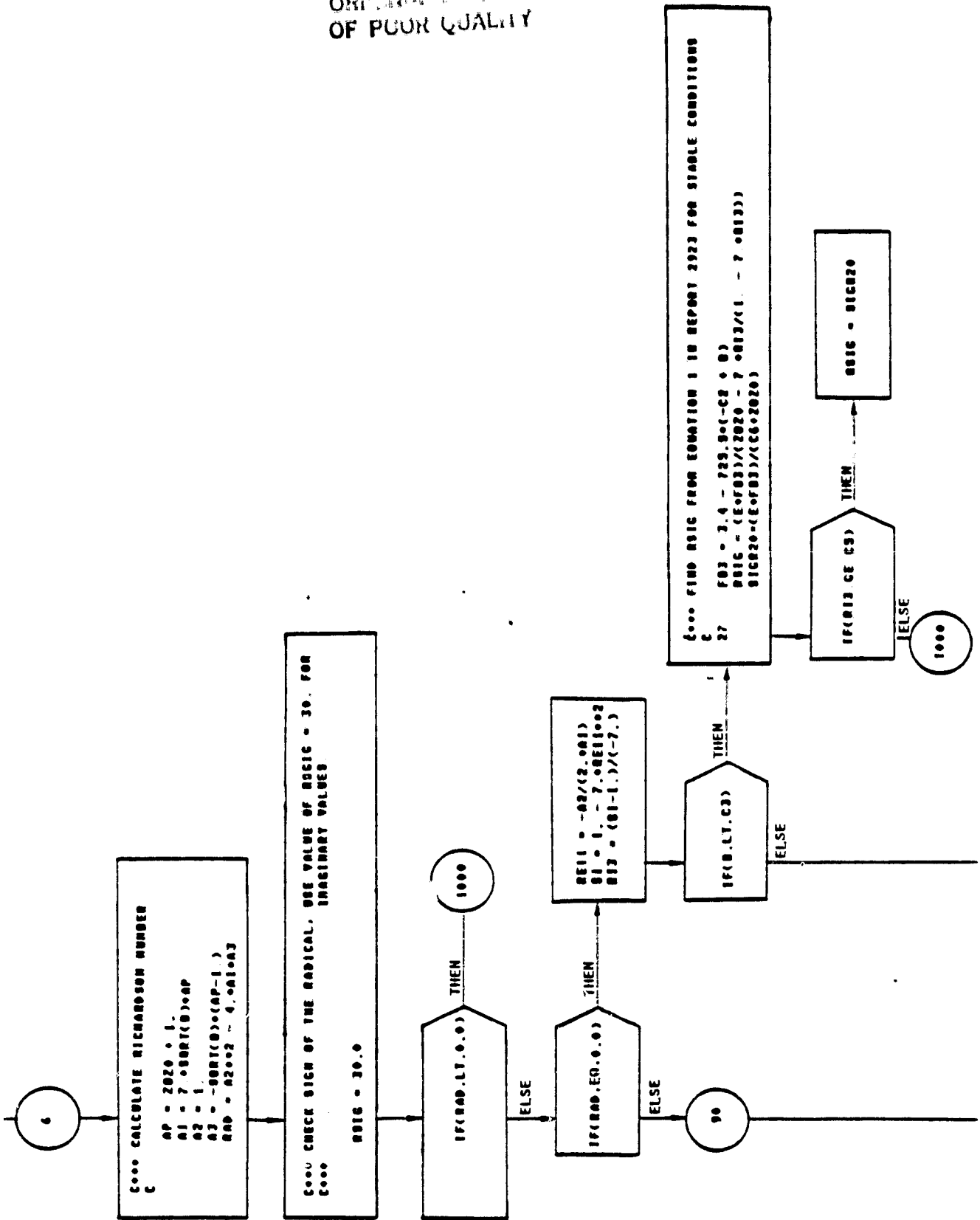


Figure 2-5. Detailed Structure Flow Diagram of Subroutine RSGAZ (continued)



ORIGINAL PAGE IS  
OF POOR QUALITY

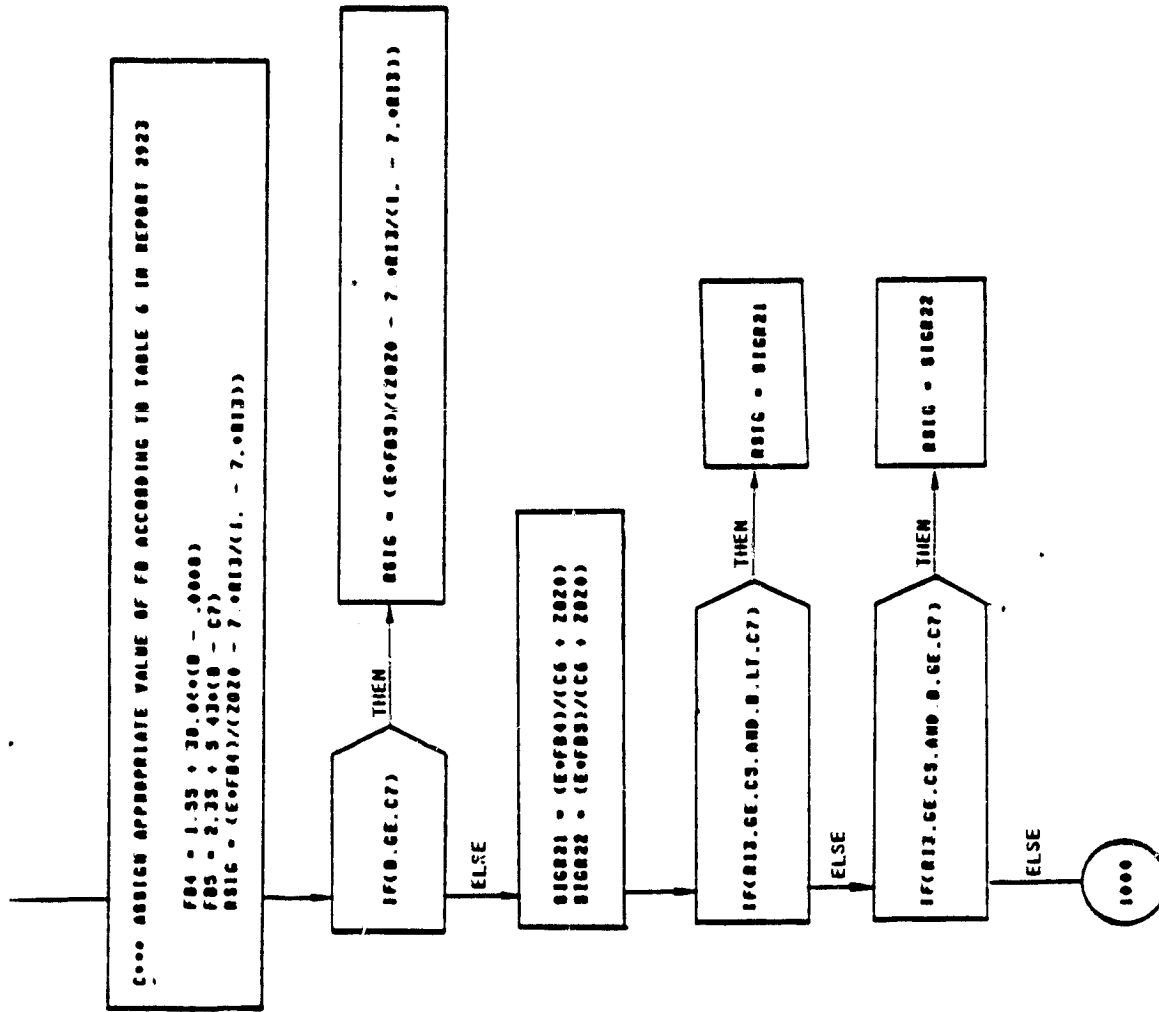


Figure 2-5. Detailed Structure Flow Diagram of Subroutine RSGAZ (continued)



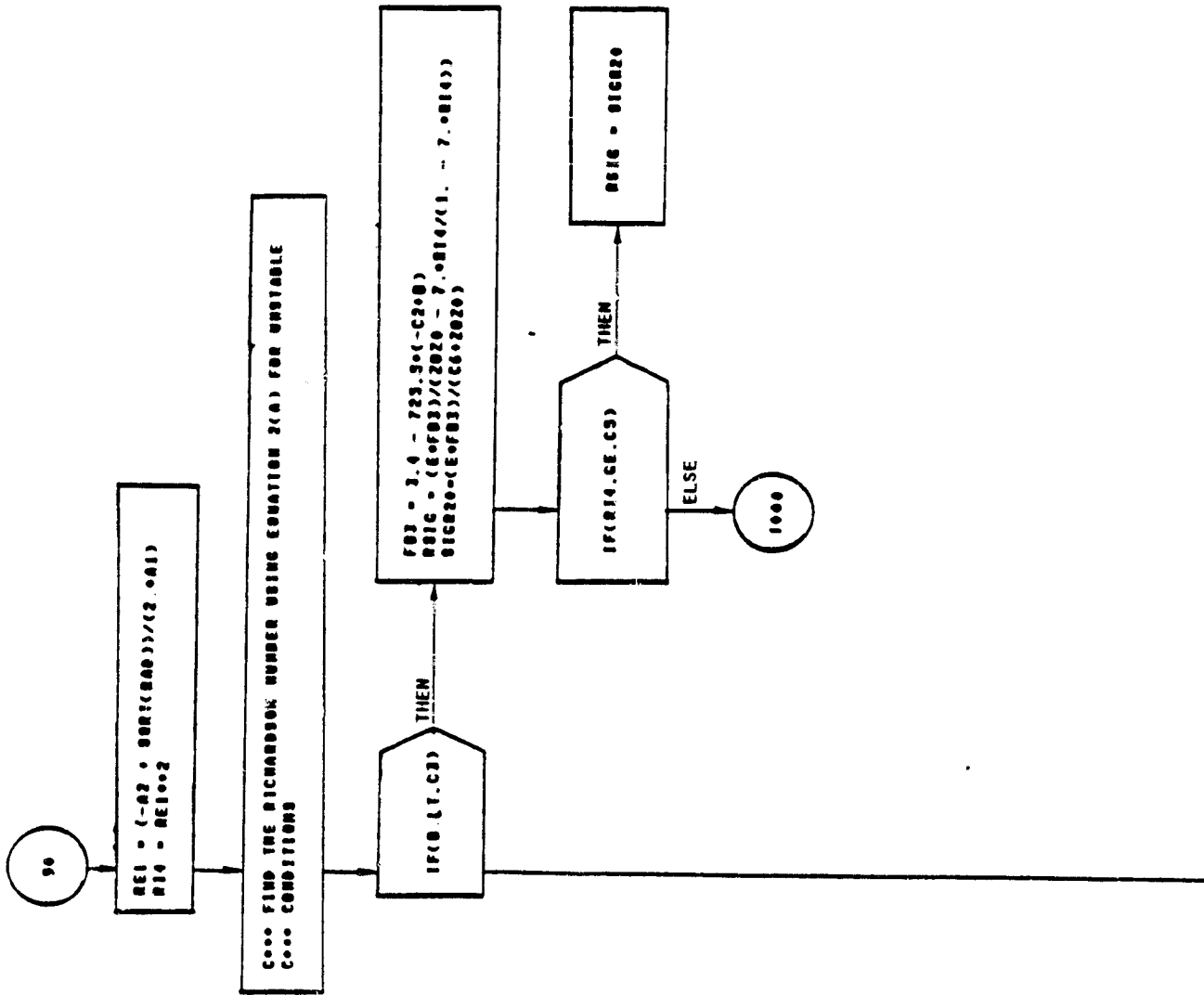


Figure 2-5. Detailed Structure Flow Diagram of Subroutine RSGAZ (continued)



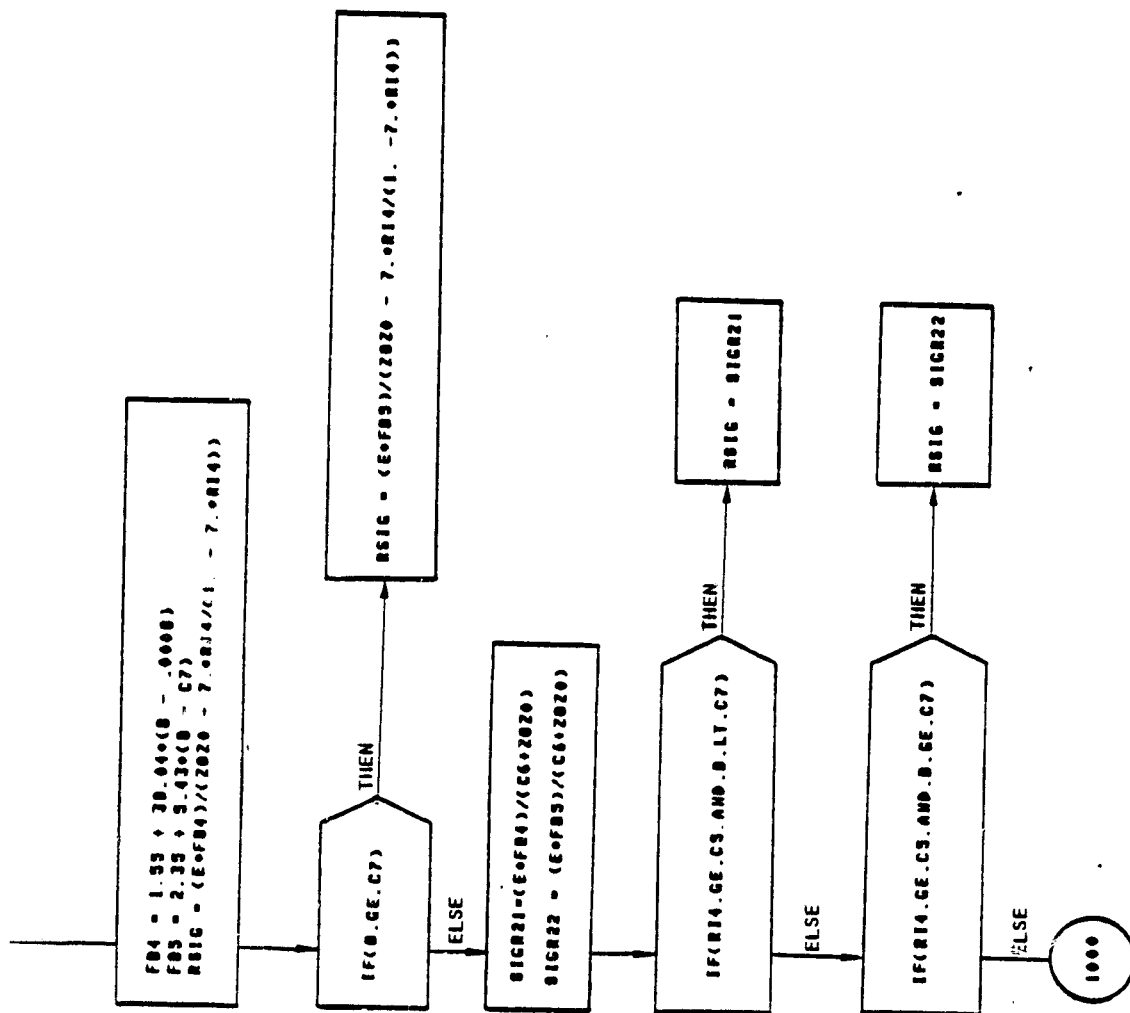


Figure 2-5. Detailed Structure Flow Diagram of Subroutine RSGAZ (continued)



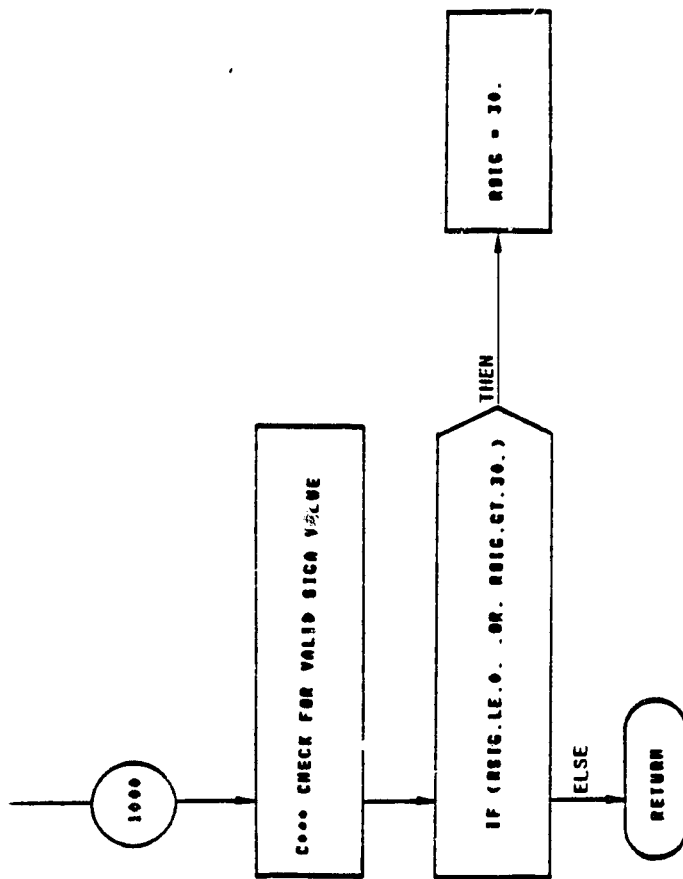


Figure 2-5. Detailed Structure Flow Diagram of Subroutine RSGAZ (concluded)



selected (via entry of an integer > 8) an interactive selection of the layer height is assumed. In either case, the common data are written to "REEDS," and the meteorological data plotting routine "RMTPS" is called. Upon conclusion of RMTPS, control is returned to RCLDS. If met plots only were desired, the call to "RMTPS" is followed by a program STOP for immediate termination of execution. Otherwise, the height selected during "RMTPS" becomes the estimated mixing layer height for the balance of the calculations.

Contaminant Source Strength. Calculation of the mixing layer height determines the source strength of HCL (from which the other contaminant strengths are determined) in that it determines the maximum altitude over which the rocket vehicle effluent production algorithm may operate; contaminants dumped above the surface mixing layer are not permitted to diffuse into the layer.

Final Transport Speeds Direction, Wind Azimuth Uncertainty and Stabilization. Definition of the mixing layer height also permits correct averaging of transport speed and direction over the layers within the mixing layer, so this is accomplished and the results written on the selected printing device. The user is then given the chance to change the wind azimuth uncertainty, if the RESEARCH or OPERATIONAL mode is active. If not "N" for No, the user enters the uncertainty desired in wind azimuth (degrees) and  $\sigma_{AZ}$  is printed on the printing device. Since the mixing layer height selected ( $H_T$ ) may have resulted in part of the cloud being above the mixing layer top, and therefore being discarded, the "effective" stabilization height ( $H_{SE}$ ) of a "revised" cloud containing only that portion within the mixing layer must be calculated. This is done according to

$$H_{SE} = \begin{cases} \frac{H_T + H_S - R_C}{2}, & H_T - H_S \leq R_C \\ \frac{H_T}{2}, & |H_T - H_S| > R_C \end{cases}$$

where  $H_S$  is the stabilization height and  $R_C$  the cloud radius.





Similarly, the vertical extent from cloud center ( $\sigma_{z0}$ ) is calculated as

$$\sigma_{z0} = \frac{H_T - H_S + R_C}{4.3}, \quad H_S + R_C \geq H_T$$

$$\sigma_{z0} > 0$$

$$= \sigma_{X0} = \frac{\gamma_M H_S}{4.30}, \quad H_S + R_C < H_T$$

$$\text{or } \sigma_{z0} \leq 0$$

where  $\sigma_{X0}$  is the initial cloud radius.

The stabilization height is written to the printing device, the common data are written to the disk file, and the effluent concentration module "RCONS" is called. Following this, a program STOP is generated, returning control to the "REEDS" executive module.

### 2.1.3 The RMTPS Meteorological Data Plotting Module

The "RMTPS" module generates a plot of the Meteorological Profile Variables of wind speed and wind direction as a function of altitude, and the initial and effective stabilized cloud shape. The mixing layer is also indicated on the plot and can be interactively determined from user input. As an additional feature, the sounding (or other) sampling points are indicated as additional tic marks on the ordinate altitude scale. The "RMTPS" module also will draw the met plot forms if the user has selected that option in "REEDS."

Forms Generation. After the common disk file is read to load the common data, the RMTPS module checks for the forms switch, MFORM. If on (MFORM = 1), the form is generated; the first two hundred active lines (or so) are dedicated to this purpose.

Met Plots. After the forms are generated, or when forms generation is bypassed, the MONLY common parameter (OPTION control pseudo-block) is checked to see if MET plots are skipped altogether (or in a forms generation run or in a repeat run designed to prevent needless replotting of the MET-related data). The wind direction data is rotated to conform to the preset site orientation, and is scaled to plot units. The launch data and time, site data, surface pressure and



density, stabilization height and wind azimuth uncertainty are all encoded into ASCII character strings and written by the plotter onto the MET plot. This is followed by the dry temperature, potential temperature, wind speed, and wind direction (all surface values), which are likewise encoded and written by the plotter. In each case, the encoding is accomplished by the combination of "CODE" calls and subsequent WRITE to the character string, IALPHA. The character string is then written at the designated plotter location via the "CHAR" subroutine.

Curve Drawing. Actual curve drawing then proceeds by pulling the array values for the various arrays out and using them as data for the "CHAR" calls. This results in the indicated symbol from the symbol array ICRVT being plotted. In the case of the line plots, either "LINE" (for solid lines) or "OLINE" for dashed lines are used for curve generation on the plotter.

Axis/Tic Marks. Curve drawing is then followed by axis generation. This is accomplished with simpler calls to LINE, whether for axes or for tic marks.

Cloud Drawing. The cloud schematic is drawn by the "CLOUD" subroutine in the lower right quadrant of the plot. Relative distances with respect to mixing layer are kept to approximate scales.

Mixing Layer Height. At the appropriate height on the altitude scale (ordinate), the estimated mixing layer is drawn with the "LINE" routine. The "MOVEM" subroutine is then called to prompt the user to accept or replace the indicated height. The choices are:

"CO"	for continue
"UP"	for move up one met level
"DO"	for move down one met level

After each prompting, the new line is drawn in. When it is finally accepted by entering "CO," the line is labeled by the encoded layer height (meters).

Effective Cloud. Some cloud parameters (radius, height, etc.), calculated in RCLDS but not saved, are recalculated. This permits redrawing the cloud schematic with the "effective" parameter. This time, however, the "LINE" routine is used with the cloud extent data to produce a shaded ellipse superimposed over the original cloud outline. After this operation, the common data are written to the disk file and a program "STOP" is generated, returning control to "RCLDS."



#### 2.1.4 The RCONS Centerline Concentration Module

The RCONS concentration module computes the range and azimuth of the cloud center relative to the pad, calculates and plots the centerline dose and concentration of the major hazardous effluents, computes the off-centerline dose and concentration for specific locations, and calls the "RISOS" dosage and concentration isopleth plotting module. The "RCONS"/"RISOS" combination comprises the "kinematic phase" of the effluent diffusion problem. Refer to Figure 2-3 for the RCONS top-level flow.

The turbulent diffusion problem may be described by the following diffusion equation:

$$\frac{\partial \chi(\bar{r}, t)}{\partial t} + \langle \bar{v} \rangle \cdot \nabla \chi(\bar{r}, t) = \nabla \cdot \left[ \tilde{K}(\bar{r}, t, p, T) \cdot \nabla \chi(\bar{r}, t) \right]$$

where  $\chi(\bar{r}, t)$  is the scalar concentration of the diffusing effluents,  $\langle \bar{v} \rangle$  the expected value of the wind velocity and  $\tilde{K}(\bar{r}, t, p, T)$  the diffusion diagonal tensor which is a function of temperature, pressure, position and time. The separation of the problem into two (2) distinct phases, a thermodynamic phase resulting in approximate equilibrium with the ambient environment and a kinematic phase, with diffusion from the thermodynamically stable state, facilitates the linearization of the equation and subsequent separation of variables solution. We have already looked at the first phase (RCLDS), and now wish to examine the latter.

The first phase left the exhaust cloud initially spherical in shape with the cloud center at a stabilized altitude. Then, the revision of the cloud occurred to account for surface mixing layer effects. Now, the ellipsoidal expansion of the revised cloud occurs. The generalized dosage equation driving this expansion is composed of four terms:

$$\text{DOSAGE} = \left( \begin{array}{c} \text{PEAK} \\ \text{DOSAGE} \end{array} \right) \times \left( \begin{array}{c} \text{LATERAL} \\ \text{TERM} \end{array} \right) \times \left( \begin{array}{c} \text{VERTICAL} \\ \text{TERM} \end{array} \right) \times \left( \begin{array}{c} \text{DEPLETION} \\ \text{TERM} \end{array} \right)$$

In subsequent modeling, the depletion of material from the cloud by gravitational settling and rain scavenging is included, but in the current REEDS model, no



depletion of this type is included. Therefore, the driving dosage algorithm is:

$$\begin{aligned}
 D \{ x, y, z_B < z < z_T \} = & \frac{Q_M}{2\pi \sigma_y \sigma_z \bar{u}} \left\{ \exp \left[ \frac{-y^2}{2\sigma_y^2} \right] \right\} \left( \exp \left[ \frac{-(H-z)^2}{2\sigma_z^2} \right] \right. \\
 + \exp \left[ \frac{-(H-2z_B+z)^2}{2\sigma_z^2} \right] + \sum_{i=1}^{\infty} & \left\{ \exp \left[ \frac{-[2i(z_T-z_B) - (H-2z_B+z)]^2}{2\sigma_z^2} \right] \right. \\
 + \exp \left[ \frac{-[2i(z_T-z_B) + (H-z)]^2}{2\sigma_z^2} \right] + \exp & \left[ \frac{-[2i(z_T-z_B) - (H-z)]^2}{2\sigma_z^2} \right] \\
 + \exp \left[ \frac{-[2i(z_T-z_B) + (H-2z_B+z)]^2}{2\sigma_z^2} \right] & \left. \right\} \left. \right)
 \end{aligned}$$

where:

$Q_M$  = Source Strength (total mass in the indicated surface layer)

T,B subscripts indicate top and bottom of stabilized cloud, respectively.

H = Height of Cloud Centroid

Z = Height within layer ( $z_B \leq z \leq z_T$ )

$\sigma_y, \sigma_z$  = Standard Deviation of the Time Dependent Cloud Dimensions in the y and z directions, respectively.

Centerline values are obtained at  $y = 0$ , while surface values are obtained at  $z = 0$ . From the dosage, the concentration is formed by the expanding front as

$$\chi(x,y,z,t) = \frac{D(x,y,z,t) \bar{U}}{2\pi \sigma_x}$$



where  $\bar{U}$  is the average wind speed, and  $\sigma_x$  is the along-wind cloud dimension. The balance of the kinematic phase modules are concerned with establishing positions and timings relative to the cloud center and time of maximum dosage/concentration, and the relative distribution of dosage/concentrations among the effluent constituents.

Centerline Dosages and Concentrations. After some input terminal initialization, the standard common disk file read, and some useful parameter initialization, the range down the cloud centerline is computed and expressed relative to the pad. The dosage algorithm "guts" exists in subroutine DFEXP, where the various sums in the above equation are computed. DFEXP is called with  $y = 0$  off centerline distance to obtain the centerline maximum dose and concentration. The arguments of DFEXP are the surface mixing layer height and the concentration desired. DFEXP can then obtain the appropriate dimensions corresponding to that concentration. However, the specification of the number 1000 for concentration is a key that causes DFEXP to produce only values at the indicated range, and report the standard deviation of the cloud dimensions in the transverse direction.

Cloud Length and Arrival/Departure Time. Based on the calculations of cloud extent made by DFEXP, the arrival and departure times at the indicated range are computed and printed with the range and azimuth relative to the pad, at 250 meter increments along the cloud centerline.

RCONS User Prompts. Most prompts for RCONS user action are stored in the RCONS question file "?RCONS," and are accessed via the DREAD subroutine.

Centerline Plots. If the current operational mode is not production, the user is then prompted to decide whether or not to produce actual plots of the centerline dosage and concentration. If the response is not "N" for no, subroutine CPLOT is called with the centerline range array generated above. The plot is generated with a pre-set scale factor in order to coincide with traditionally generated forms. The default response to the centerline plot prompt is "YES."

Maximum Dosage/Concentration. The maximum dosage and concentration will occur along the cloud centerline. The point at ground level where this occurs had been calculated and saved during the above centerline dosage/concen



tration calculation. They are now written out on the printing device selected with pad-relative range and azimuth.

Off-Centerline Concentrations/Dosages. If the current operating mode is PRODUCTION, the run is effectively complete and transfer of control is to the end of RCONS, and ultimately to the REEDS executive module for termination or restart. If the current mode is not PRODUCTION, however, the user is prompted for "off-centerline dosages/concentrations." If the answer is not "N" for "No," off-centerline concentrations/dosages are assumed to be desired (which is the default case), and the ORIGIN subroutine is called to set the origin for the desired launch pad and map. Application of the REEDS code to other sites can be accomplished by appropriately modifying the vehicle/pad association data in the ORIGIN subroutine. From ORIGIN, the user is prompted for the pad number and land/sea map combination. The choices are listed by the prompt, depending on the vehicle selected during REEDS input (certain vehicles imply certain pads). The user then inputs the code listed in the prompt which corresponds to the pad/map combination he desires. The user will now be prompted for pad-relative off-centerline ranges and azimuth. However, if the operating mode is RESEARCH, the user is first prompted (from the program) to enter a cloud centerline (stabilization-relative) range. This is then transformed to a pad-relative range and azimuth using the setup from ORIGIN. This is an aid in the verification of data from the centerline calculations and/or an additional centerline/off-centerline consistency check. In either RESEARCH or OPERATIONAL modes, the user is then prompted for the pad-relative range followed by the pad-relative azimuth. The entered range and azimuth are printed on the terminal display, and then on the printing device. The coordinates are plotted on the Isopleth plotting map requested by ORIGIN according to the setup resulting from the requested pad/map combination. The point is labeled by the ordinate number of the range/azimuth pair (first entry = 1, second entry = 2, etc.). Two additional points are then computed at the requested range, but the first at  $10^0$  less azimuth and the second at  $10^0$  more azimuth. All three range/azimuth pairs are then transformed to cloud center-relative coordinates, and DFEXP is called. The concentration and dosage is evaluated at each point and the  $\pm 10^0$  data used to produce an uncertainty in the concentration/dosage for each effluent. The resulting data is printed on the printing device. The format is:



MATERIAL	CONCENTRATION (PPM)	$\pm$	CONCENTRATION UNCERTAINTY	DOSAGE (PPM-SEC)	$\pm$	DOSAGE UNCERTAINTY (PPM-SEC)
----------	------------------------	-------	------------------------------	---------------------	-------	------------------------------------

The above off-centerline calculation is repeated until a zero (0) range is entered.

Isopleth Plotting. The user is then prompted for Isopleth plotting. If the answer is not "N" for "No," the Isopleth plotting module is scheduled, otherwise RCONS is terminated and control returns to REEDS through RCLDS, for program termination/restart.

### 2.1.5 The RISOS Isopleth Plotting Module

The RISOS module essentially repeats off-centerline calculations using DFEXP, except it uses a clever series of calls resulting in iso-concentration contours. Refer to Figure 2.3 for RISOS top level flow.

Isopleth Initialization. The user terminal is first initialized and the common data read from disk via subroutine BWDIS. The ORIGIN subroutine is called again in case the call in RCONS was bypassed by rejection of the off-centerline option. This makes sure the plotting origin is established relative to the vehicle pad/map configuration desired. -

Cloud Rise Ground Track. The cloud rise ground track from the pad to stabilization is then plotted by calculating the range of the cloud centroid from the time required to rise through each meteorological layer. The times for each layer were stored during the cloud rise module (RCLDS) execution. The average wind velocity in each layer is integrated over the stored time to provide the incremental range that is plotted. Since the wind velocity is measured "from" the source, the cloud transport direction must use a factor of  $\pi$  to properly orient its movement. The process is terminated when the layer containing the stabilized cloud center is reached. However, an additional segment is required, because of the use of the "effective" stabilization height, which replaced the stabilization height when placement of the surface mixing layer height severed part of the cloud (for diffusion purposes). The stabilization point is then labeled with a "+." Next, the point of maximum concentration at ground level, stored during RCONS, is labeled with an "@." Finally, the cloud track downwind from stabilization is plotted with a straight line using the transport direction from that point.



### 3.0 REEDS OPERATION

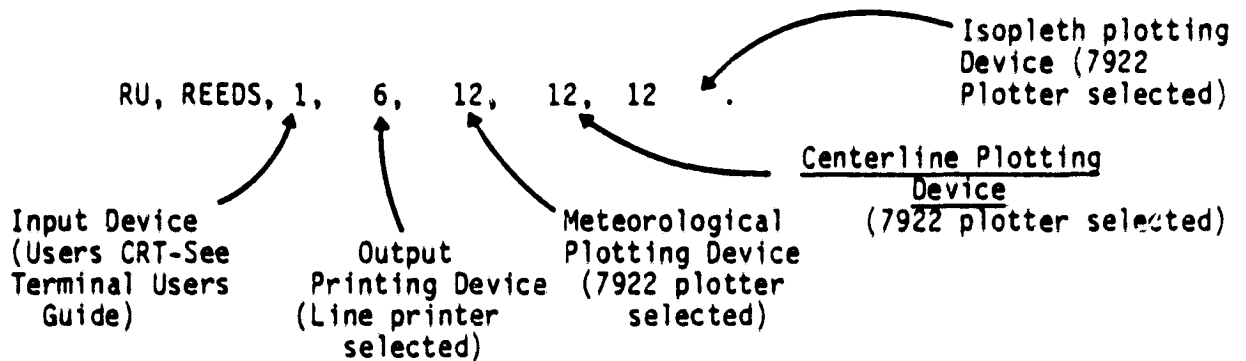
The basic form of storage for REEDS is in the form of source code files for the program/subroutine packages and ASCII data files for the input data. Ultimately, the source code files are compiled and thereby translated into machine language modules. The latter are loaded, SP'd, OF'd and RP'd (see terminal users manual). The "RU" command is then used to execute the RP'd modules while arguments are used to pass special information to the program. The compiled, loaded, and SP'd files are convenient to use, since they merely have to be RP'd prior to execution--a very quick process. One drawback is that there is little way of verifying that the RP'd module is the result of the source file as it appears. Also, since source files must be saved anyway, the existence of a large compiled file may seem redundant, especially if not used often. Nevertheless, because of the lengthy loading process, the compiled files are often stored. Thus, for purposes of showing a run, the compiled files are assumed to exist in the RP'd state. Maintenance files have been built which automatically perform the operations leading to an RP'd program. It is the purpose of this section to explain actual REEDS execution.

#### 3.1 THE TEST CASE

A test case was chosen, using the space shuttle as the vehicle and the 8 A.M. EDT 16 November 1969 rawinsonde as the meteorology. The file name of the latter data was changed to DATA01, just for convenience, and can be seen in Section 4.4 of this report. The CRT terminal used was equipped with a hardcopy device, making it possible to get intermittent copies of the screen to aid the narration.

Run Time Parameters. The REEDS system was designed to serve MSFC research needs and KSC operational needs. One difference between these is that KSC has three plotters, each one assigned to a major plot output. MSFC has one plotter for all three plots. The allocation in the run statement of printing and plotting devices to logical units makes this possible. The user enters:





At this time the screen appears as in Figure 3-1. The default responses are indicated on the CRT screen when they appear by easily noted enhanced characters.

Operational Configuration. The next five (5) entries deal with configuring the logical path through the modules. The first prompt asks about the low altitude curve fit. We choose not to use the least squares fit for the potential temperature gradient and answer "No." We have renamed the 16 Nov 1969 rawinsonde file DATA01 for this run, and therefore select the default, "YES," response for the meteorological data file name (response-2). For the third input, the default OPERATIONAL mode is chosen, followed by a similar choice for Met Plot Form rejection (4th) and Met plot creation (5th). The printer output default ("Yes") is selected, and the screen appears as in Figure 3-2.

Launch Specific Parameters. The default launch time is the current time on the computer's system clock. It was decided to select the time rather than enter either a realistic or other time. Likewise, the default vehicle (space shuttle) is chosen as the launch vehicle. At this point, the screen appears as in Figure 3-3.

Initial Printed Output. At this point, the initial printer data becomes available, printing the header and meteorological sounding data as shown in Figure 3-4. The printout reflects all input entered to date, except the Met. Plot Form/Met. Plot configurational data.

Estimated Mixing Layer Height. The RCLDS cloud rise module has estimated the height of the top of the surface mixing layer, printing 2552 meters on the screen. The user is then prompted to raise or lower this height. The cloud rise has been calculated and stored up to this height, and RMTPS has been called.



---

RU,REEDS,1,6,12,12,12

\*\*\*\*\*NASA/MSFC EXPERIMENTAL MULTILAYER DIFFU15 AUG 1981\*\*\*\*

---

Figure 3-1. Code Initiation

---

\*\*\*\*\*NASA/MSFC EXPERIMENTAL MULTILAYER DIFFU15 AUG 1981\*\*\*\*

DO YOU WANT THE LOW ALT. CURVE FIT?	YES	NO	NO
LOW ALTITUDE CURVE FIT			NO
USE DEFAULT COMMON DATA FILE (DATA01)	?	YES	
RUN TYPE:	OPERATIONAL		
GENERATE THE MET PLOT FORM?			NO
MET PLOTS?		YES	
GENERATE PRINTER OUTPUT?		YES	
DEFAULT LAUNCH TIME/DATE?	1833 CDT 15 AUG 1981	YES	NO

---

Figure 3-2. Configurational Paths Added



---

\*\*\*\*\*NASA/MSFC EXPERIMENTAL MULTILAYER DIFFU15 AUG 1981\*\*\*\*  
DO YOU WANT THE LOW ALT. CURVE FIT? YES NO NO  
LOW ALTITUDE CURVE FIT NO  
USE DEFAULT COMMON DATA FILE (DATA01) ? YES  
RUN TYPE: OPERATIONAL  
GENERATE THE MET PLOT FORM? NO  
MET PLOTS? YES  
GENERATE PRINTER OUTPUT? YES  
DEFAULT LAUNCH TIME/DATE? 1833 CDT 15 AUG 1981 YES NO  
SELECT LAUNCH VEHICLE: SHUTTLE=S  
ESTIMATED TOP OF SURFACE LAYER(M) IS: 2552.4  
  
MOVE TOP OF SURFACE LAYER: UP DOWN CONTINUE

---

Figure 3-3. Launch Specific Data Added



Initial Plotter Output. The meteorological data are plotted, since Met. Plots were selected; however, the form is omitted because a blank form already exists and is used. The blank form appears in Figure 3-5. After the Meteorological data are plotted, the plot appears as in Figure 3-6. The estimated mixing layer top is drawn, but not labeled, and the meteorological sounding data points are indicated by the additional irregularly spaced tic marks on the left ordinate. The program is left in an input mode, awaiting instructions as to mixing layer height modification.

Mixing Layer Height Modification. As a demonstration of the program options rather than a reflection of reality, the cloud is shown stabilized at 871 meters, and the mixing layer height is dropped four (4) times to 1106 meters. As can be seen in Figure 3-7, this is accomplished by entering four (4) consecutive "DO"'s for (DOWN) and finally a "CO" (for continue).

Final Cloud Rise. With the mixing layer finalized, the cloud rise is modified to reflect "severing" of part of the cloud by the boundary and subsequent recalculation of the stabilization height (to become the "effective" height) and total rise time. The resulting "effective" cloud is shown in Figure 3-8, and the cloud rise summary data printed on the printer is shown in Figure 3-9.

Final Wind Azimuth Uncertainty. The wind azimuth uncertainty is calculated from the mixing layer height and printed on the terminal display (Figure 3-7). The user is given a chance to modify this value. As the figure shows, the estimated value was accepted.

Centerline Concentrations and Dosages. The table of calculated centerline HCL dosages and concentrations are printed as shown in Figure 3-10. The user is prompted for centerline plots of concentration and dosages. These are desired (Figure 3-11) so a blank centerline form is loaded onto the plotter (Figure 3-12). The centerline dosage and concentration are plotted on the plotter, and the plot is labeled (Figure 3-13). The maximum ground concentration and dosage are then printed on the printer, as are the pad-relative coordinate (Figure 3-14). All dosages and concentrations are HCL dosages and concentrations. All other effluents are determined from HCL, the chief effluent.



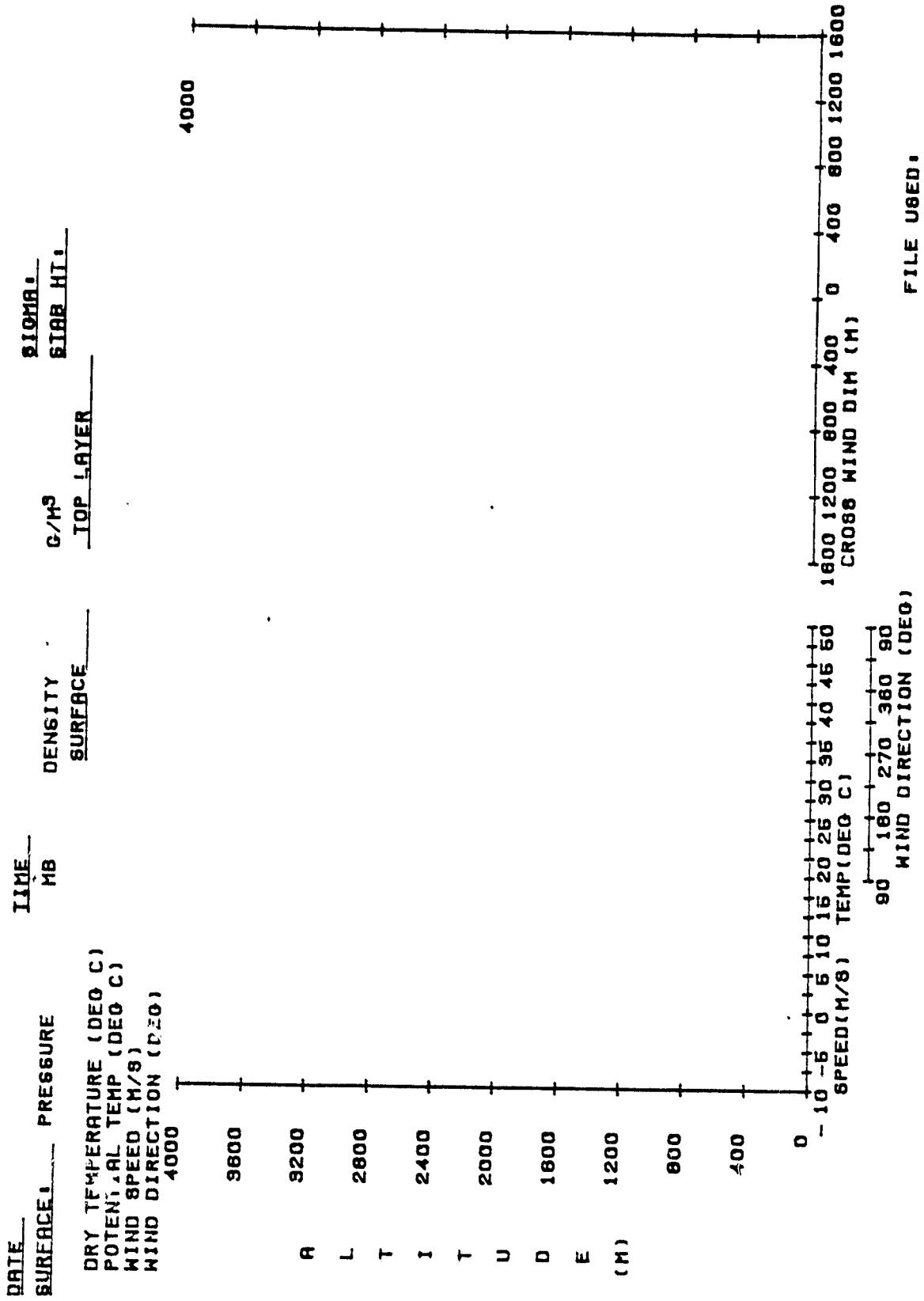
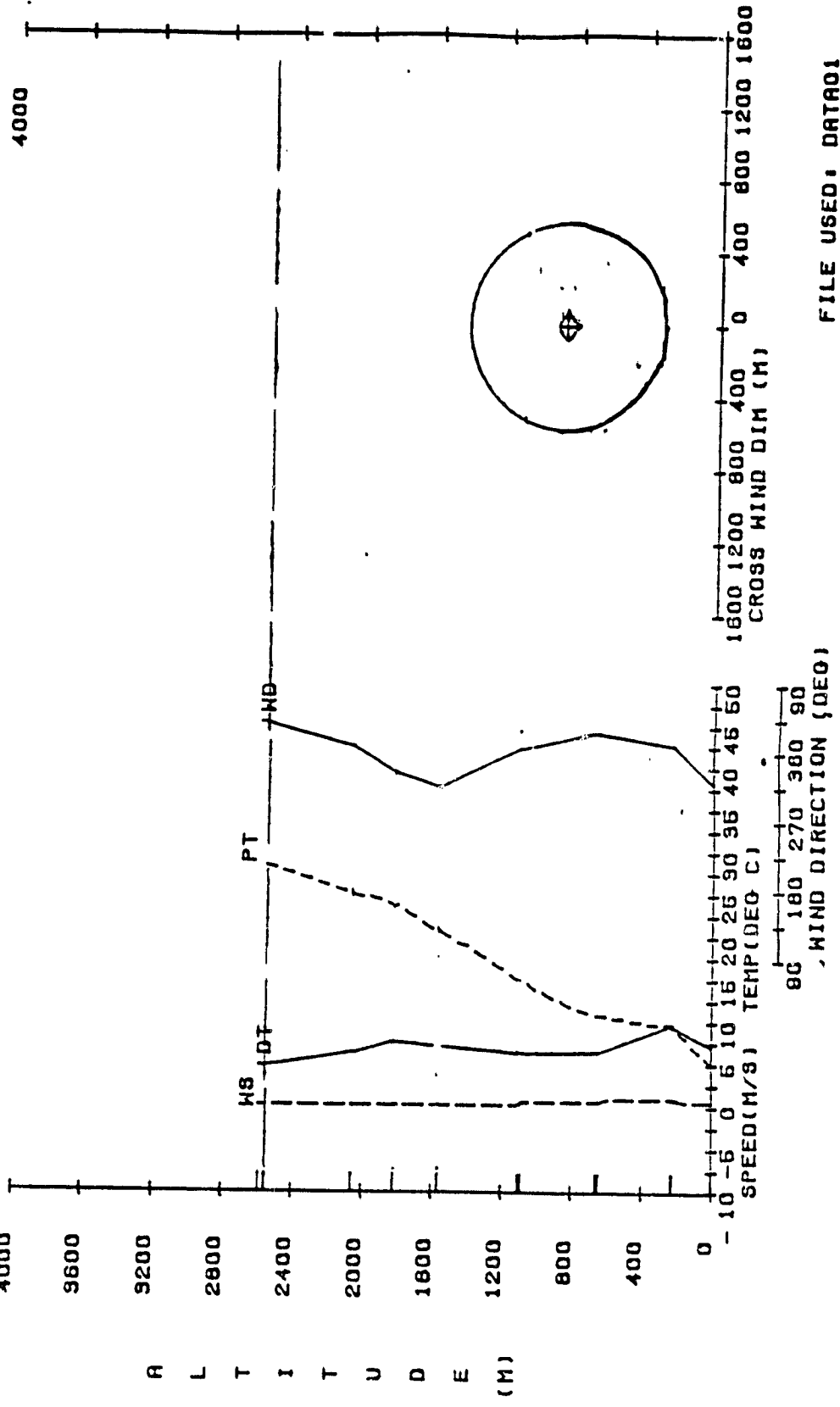


Figure 3-5. Blank Meteorological Form



DATE 16 NOV 1968 TIME 000 EDT 010MA 90.0  
 SURFACE PRESSURE 1027.0 MB DENSITY 1273.0 G/M<sup>3</sup> KSC 071.1 M  
 SURFACE 7.1 TOP LAYER  
 DRY TEMPERATURE (DEG C) 6.3  
 POTENTIAL TEMP (DEG C) 16.0  
 WIND SPEED (M/S) .9  
 WIND DIRECTION (DEG) 920.0 8.0



FILE USED: DATA01

Figure 3-6. Meteorological and Initial Cloud Data Plotted with Estimated Mixing Layer Height



---

DO

DO  
DO  
DO  
CO

CALCULATED TOP OF SURFACE TRANSPORT LAYER IS: 1106.12  
USE A SIGMA FOR WIND AZIMUTH ANGLE OF: 30.0 YES NO  
USE A SIGMA FOR WIND AZIMUTH ANGLE OF: 30.0 YES  
CENTERLINE CONCENTRATION PLOT DESIRED ? YES NO

---

Figure 3-7. Modifying the Mixing Layer Height





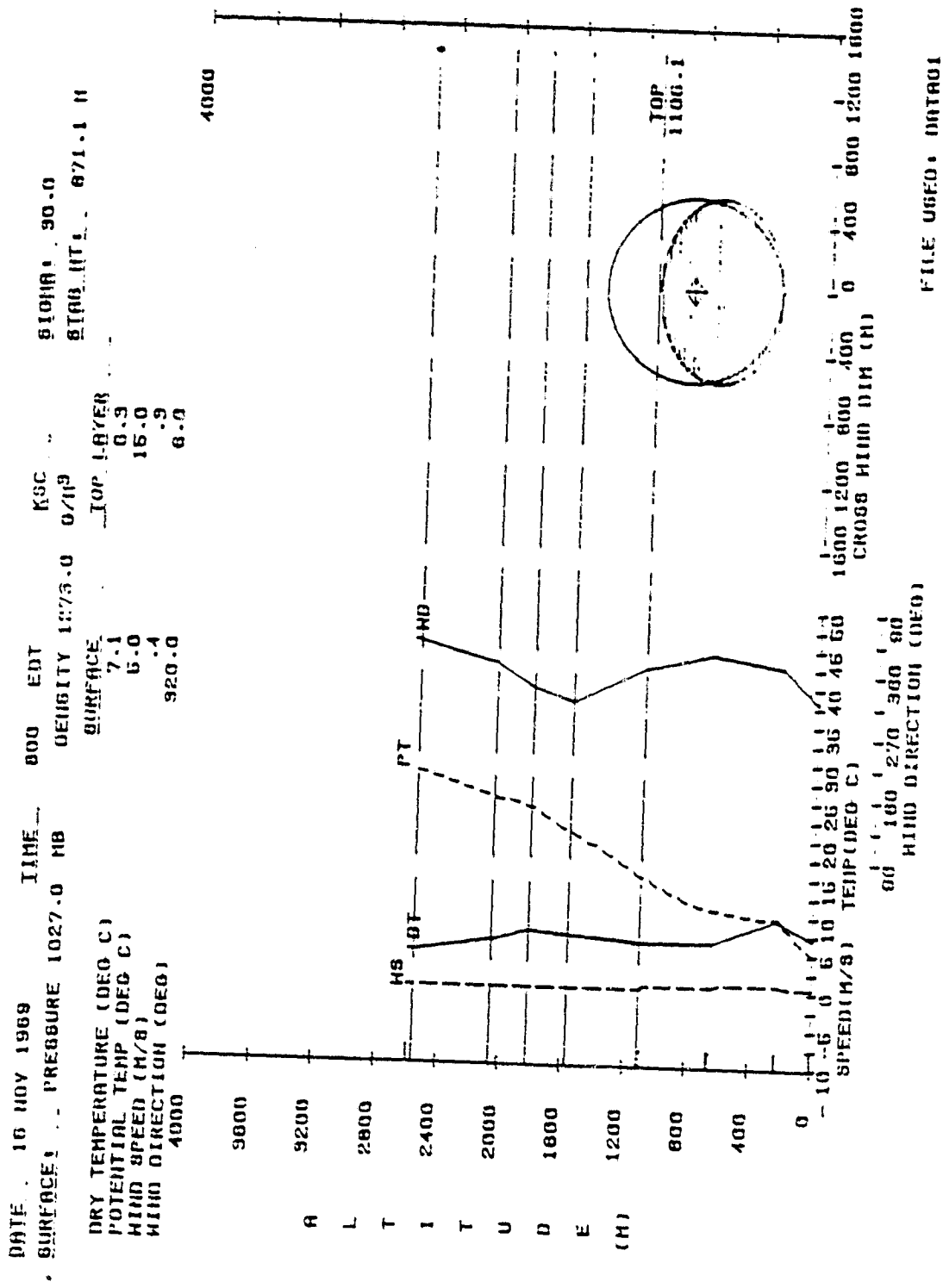


Figure 3-8. Modified Cloud - Showing Mixing Layer Height Effects



EXHAUST CLOUD

LEVEL	ALTITUDE (METERS)		RISE TIME (SECONDS)	RANGE (METERS)	DIRECTION (DEGREES)
2	224.0	STABLE	7.6	4.7	165.0
3	229.8	STABLE	8.0	5.0	166.5
4	648.0	STABLE	53.3	39.3	194.6
5	662.0	STABLE	55.5	40.8	195.1

\*\*\*\*CLOUD STABILIZATION\*\*\*\*

HEIGHT(M): 1004.2  
 STABILIZATION TIME AFTER LAUNCH(SEC): 174.0  
 RANGE FROM PAD(M): 111.7  
 DIRECTION FROM PAD(DEG): 195.3  
 PRESSURES ARE: 1027.00 1000.00 999.30  
 TBHT = .733812E+03

\*\*\*\*TOP OF SURFACE LAYER METEOROLOGICAL PARAMETERS\*\*\*\*

HEIGHT(M): 1106.1  
 WIND DIRECTION(DEG): 6  
 WIND SPEED(M/SEC): .3

\*\*\*\*DIFFUSION PARAMETERS\*\*\*\*

MEAN SPEED(M/SEC): .7  
 MEAN TRANSPORT DIRECTION(DEG): 191.3

SIGMA OF WIND AZIMUTH ANGLE, SIGAZ: 30.0

EFFECTIVE CLOUD HEIGHT (M): 733.8

Figure 3-9. Final Exhaust Cloud Rise Data



FOR QUALITY

CLOUD CONCENTRATIONS AND DOSAGES

DISTANCE (METERS)	CONCENTRATION (PPM)	DOSAGE (PPM SEC)	TIME AFTER LAUNCH(SEC)		POSITION FROM PAD	
			START	FINISH	RANGE	AZ
0.0	.000	.057	-1787	2135.	111.7	195.3
1000.0	3.831	4379.79	-261.	3661.	1111.4	191.7
2000.0	5.126	5859.96	1264.	5186.	2111.4	191.6
3000.0	4.432	5066.42	2790.	6712.	3111.4	191.5
4000.0	3.618	4135.67	4316.	8239.	4111.4	191.5
5000.0	3.005	3435.43	5841.	9763.	5111.4	191.4
6000.0	2.564	2931.11	7367.	11289.	6111.4	191.4
7000.0	2.235	2555.21	8893.	12815.	7111.4	191.4
8000.0	1.981	2264.59	10418.	14349.	8111.4	191.4
9000.0	1.779	2033.24	11944.	15866.	9111.4	191.4
10000.0	1.614	1844.72	13470.	17392.	10111.4	191.4
11000.0	1.477	1688.16	14995.	18917.	11111.4	191.4
12000.0	1.361	1556.07	16521.	20443.	12111.4	191.4
13000.0	1.262	1443.14	18047.	21969.	13111.4	191.4
14000.0	1.177	1345.48	19572.	23495.	14111.4	191.4
15000.0	1.102	1260.19	21098.	25020.	15111.4	191.4
16000.0	1.037	1185.07	22624.	26546.	16111.4	191.4
17000.0	.978	1119.39	24150.	28072.	17111.4	191.4
18000.0	.926	1058.82	25675.	29597.	18111.4	191.4
19000.0	.879	1005.27	27201.	31123.	19111.4	191.4
20000.0	.837	956.869	28727.	32649.	20111.4	191.4

Figure 3-10. Centerline Hcl Concentrations/Dosages with Cloud Arrival/Departure Times and Pad-Relative Positions



---

DO

DO  
DO  
DO  
CO

CALCULATED TOP OF SURFACE TRANSPORT LAYER IS: 1106.12  
USE A SIGMA FOR WIND AZIMUTH ANGLE OF: 30.0 YES NO  
USE A SIGMA FOR WIND AZIMUTH ANGLE OF: 30.0 YES  
CENTERLINE CONCENTRATION-PLOT DESIRED ? YES NO  
YES  
CENTERLINE CONCENTRATION PLOT DESIRED ? YES

---

Figure 3-11. Centerline Concentration/Dosage Plots Added



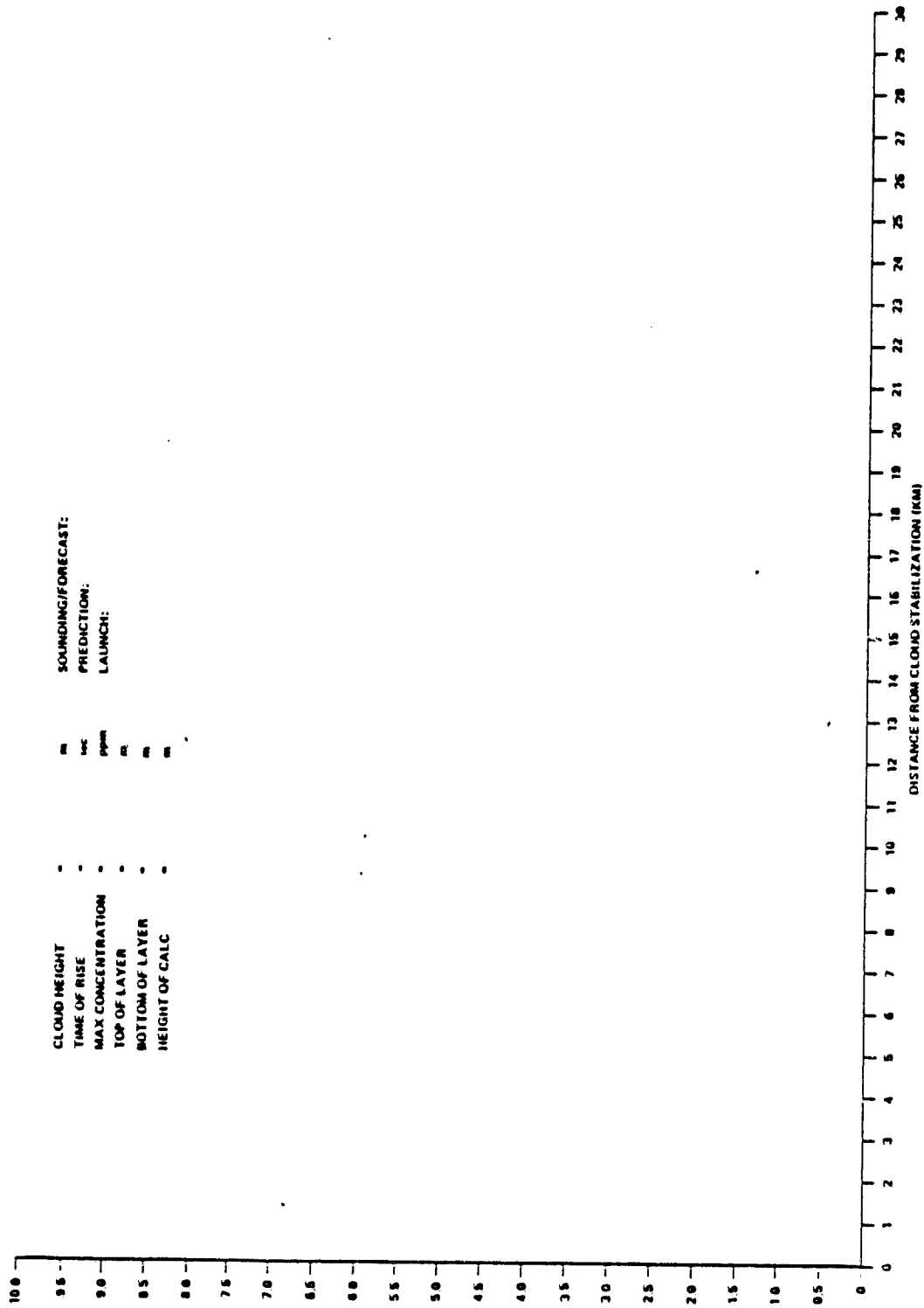


Figure 3-12. Blank Form for Centerline Concentration/Dosage Plots



SOUNDING FORECAST: 800 EDT 16 NOV 1969  
 PREDICTION 1758 CDT 15 AUG 1981  
 LAUNCH:

CLOUD HEIGHT : 734. "  
 TIME OF RISE : 174. "  
 MAX CONCENTRATION : 5.15 ppm  
 TOP OF LAYER : -1106. "  
 BOTTOM OF LAYER : 0. "  
 HEIGHT OF CALC : 0. "

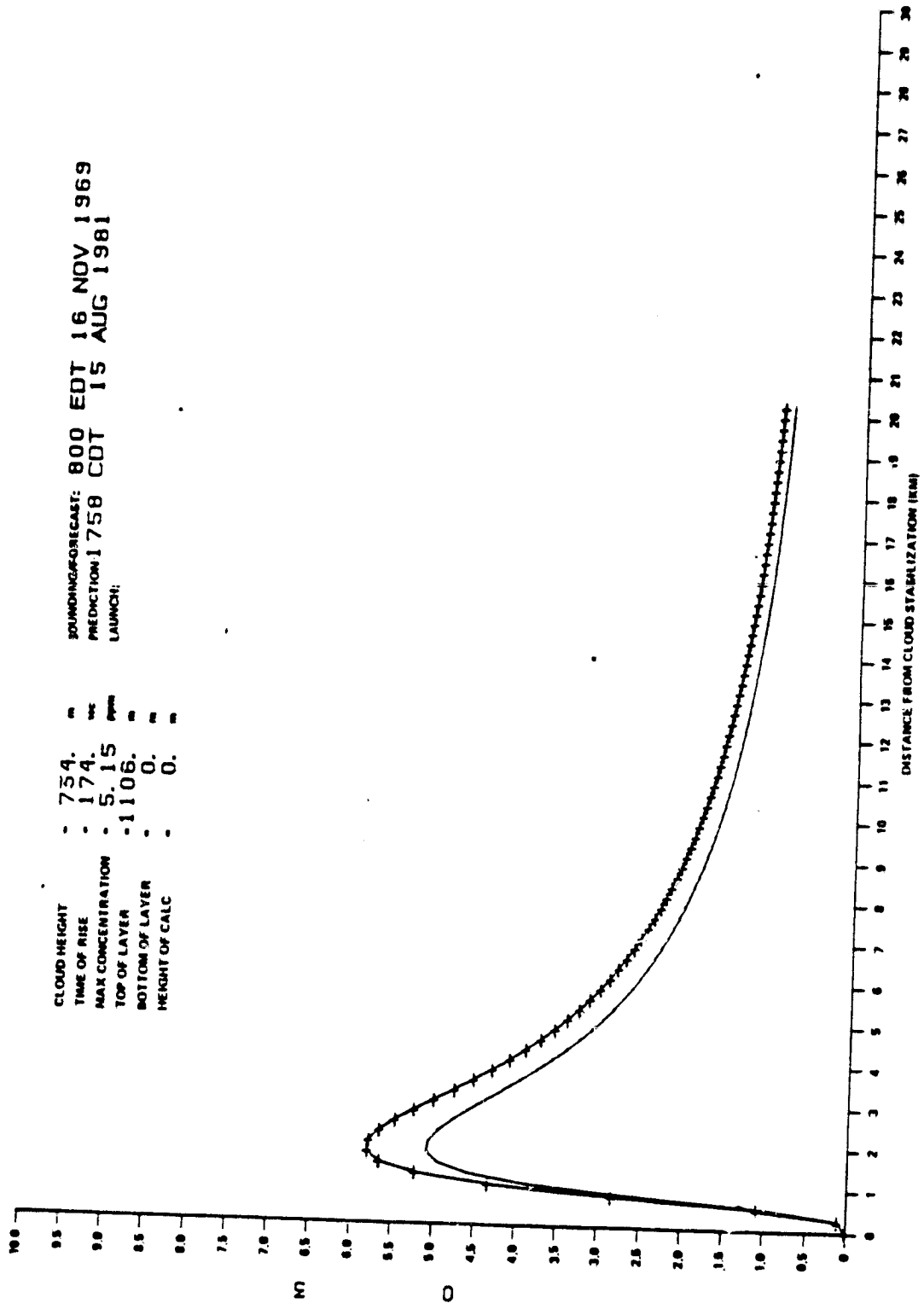


Figure 3-13. Centerline Dosage/Concentration Plots



CLOUD CONCENTRATIONS AND DOSAGES

DISTANCE (METERS)	CONCENTRATION (PPM)	DOSAGE (PPM SEC)	TIME AFTER LAUNCH(SEC)	POSITION FROM PAD RANGE AZ
0.0	.000	.057	-1787	111.7 195.3
1000.0	3.631	4379.75	-261.	111.4 191.7
2000.0	5.126	5959.96	1264.	211.4 191.6
3000.0	4.432	5066.42	2790.	311.4 191.5
4000.0	3.616	4135.67	4316.	411.4 191.5
5000.0	3.005	3435.43	5841.	511.4 191.4
6000.0	2.564	2931.11	7367.	611.4 191.4
7000.0	2.235	2555.21	8893.	711.4 191.4
8000.0	1.981	2264.59	10418	811.4 191.4
9000.0	1.779	2033.24	11944	911.4 191.4
10000.0	1.614	1844.72	13470	1011.4 191.4
11000.0	1.477	1688.16	14995	1111.4 191.4
12000.0	1.361	1556.07	16521	1211.4 191.4
13000.0	1.262	1443.14	18047	1311.4 191.4
14000.0	1.177	1345.48	19572	1411.4 191.4
15000.0	1.102	1260.19	21098	1511.4 191.4
16000.0	1.037	1185.07	22624	1611.4 191.4
17000.0	.978	1118.39	24150	1711.4 191.4
18000.0	.926	1058.82	25675	1811.4 191.4
19000.0	.875	1005.27	27201	1911.4 191.4
20000.0	.837	956.869	28727	2011.4 191.4

\*\*\*\*\*POINT OF MAXIMUM CONCENTRATION\*\*\*\*\*  
 RANGE FROM PAD(M): 1861.4  
 DIRECTION(DEC): 191.6  
 HEIGHT(M): 0.0  
 MAXIMUM CONCENTRATION(PPM): 5.145

\*\*\*\*\*POINT OF MAXIMUM DOSAGE\*\*\*\*\*  
 RANGE FROM PAD(M): 1861.4  
 DIRECTION(DEC): 191.6  
 HEIGHT(M): 0.0  
 MAXIMUM DOSAGE (PPM SEC): 5682.0

\*\*\*\*\*CONCENTRATIONS AND DOSAGES WITH 10 DEGREE UNCERTAINTIES\*\*\*\*\*

RANGE(M)	AZIMUTH(DEC)	MATERIAL	CONCENTRATION(PPM)	DOSAGE(PPM)
5000.0	170.0	HCL	.184E+01 +/- .050	2109.105 +/- 57.684
		CU	.676E-02 +/- .000	7.730 +/- .211
		CO2	.403E+01 +/- .110	4606.357 +/- 125.984
		AL2O3	.473E+01 +/- .129	5409.617 +/- 147.953
		NO	.322E-02 +/- .000	3.681 +/- .101
5000.0	150.0	HCL	.306E+01 +/- .285	3494.566 +/- 325.191
		CU	.112E-01 +/- .001	12.809 +/- 1.195

Figure 3-14. Addition of Maximum Points of Concentrations and Dosage Along Ground Track



Off-Centerline Concentration/Dosages. The user is then prompted for off-centerline concentrations/dosages. These are selected, causing a call to the ORIGIN subroutine and a further prompt for the desired pad number/map type. Of the choices listed, pad 39 was chosen, and a sea map (Figure 3-15) rather than a land map (Figure 3-16) is chosen. As will be seen, this is a mistake, as better resolution and coverage would result for a land map.

The user is then prompted for range and azimuth (pad-relative) to the point for which off-centerline calculations are desired. The prompting continues (Figure 3-17) until a zero (0) range is entered. During this time, the off-centerline effluent concentrations, their associated  $\pm 10^0$  (in azimuth) uncertainties, and dosages and their respective uncertainties, are printed on the printer (Figure 3-18). The points are also plotted on the plotter, and labeled with their ordinate index (1st one = 1, 2nd = 2, etc.) (Figure 3.19).

Isopleth Plotting. The user is then prompted for isopleth plotting. This is accepted and the default set of concentration levels is listed on the terminal display, and the user is asked if he intends to use these, or input his own (Figure 3-20). These are accepted, and the isopleths are drawn and (Figure 3-21) labeled.

Cloud Arrival/Departure Time. Finally, the cloud position and extent as a function of time are printed in a concluding table (Figure 3-22).

Termination. The user is then asked for termination ("T") or restart ("R") termination is chosen (Figure 3-20).



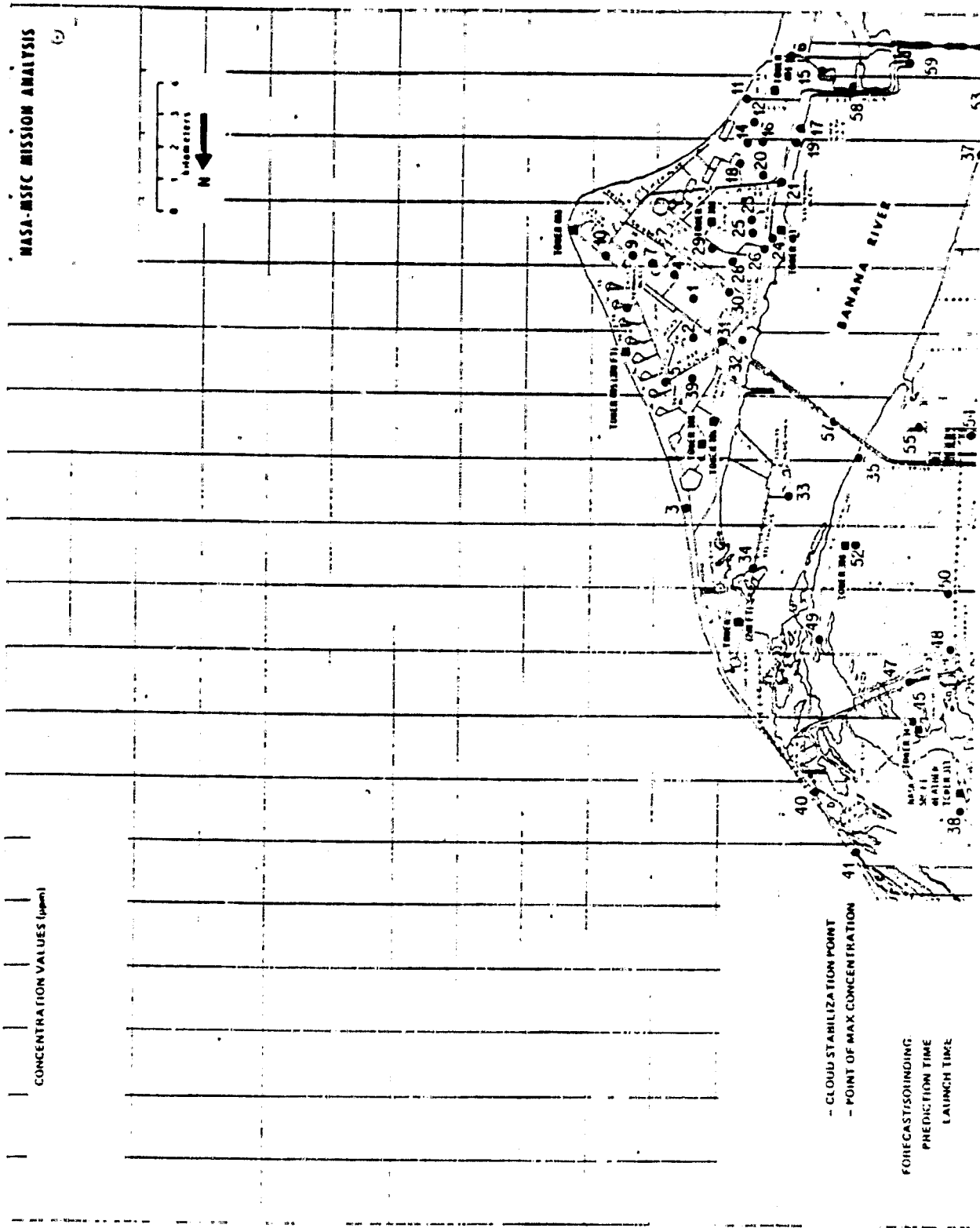


Figure 3-15. Blank Sea Map



ORIGINAL PAGE IS  
OF POOR QUALITY

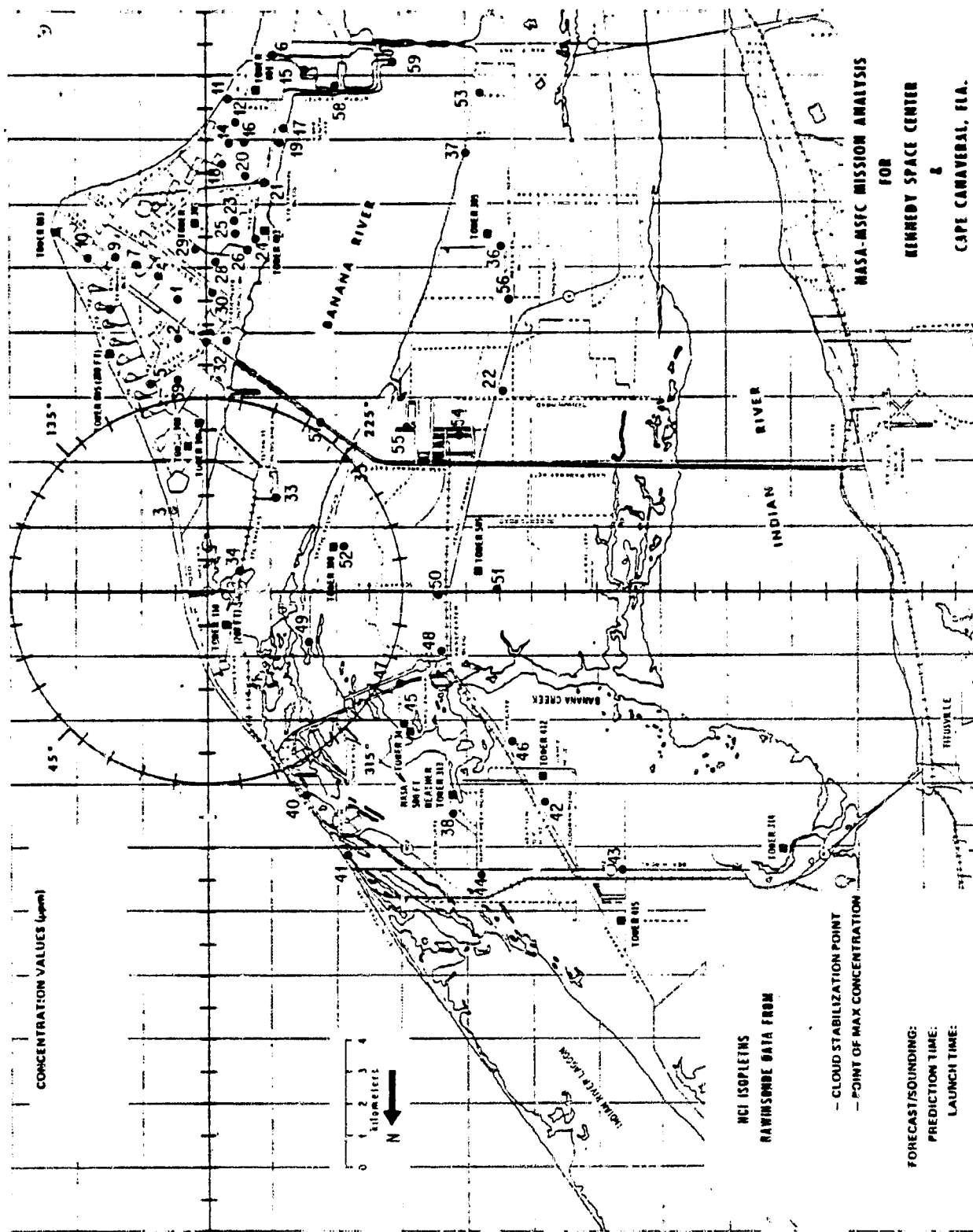


Figure 3-16. Blank Land Map



---

SPECIFIC SITE PREDICTIONS (PLOTTED)?

YES

NO

39. SEA MAP

ENTER RANGE(M) AND AZIMUTH(DEG): (Ø TERMINATES PROCEDURE)

RANGE: 1000                      AZIMUTH: 192  
RANGE:                              AZIMUTH

ENTER RANGE(M) AND AZIMUTH(DEG): (Ø TERMINATES PROCEDURE)

RANGE: 1000                      AZIMUTH: 192  
RANGE: 2500                      AZIMUTH: 192  
RANGE: 5000                      AZIMUTH: 192  
RANGE: 7000                      AZIMUTH: 192  
RANGE: 8000                      AZIMUTH: 192  
RANGE: 10000                      AZIMUTH: 192  
RANGE: 12000                      AZIMUTH: 192  
RANGE: 15000                      AZIMUTH: 192  
RANGE: 20000                      AZIMUTH: 192  
RANGE: 25000                      AZIMUTH: 192  
RANGE: 0                              AZIMUTH:

RANGE: 0                              AZIMUTH:

CONCENTRATION ISOPLETH PLOT DESIRED?

YES

NO

---

Figure 3-17. Specific Site Range/Azimuth Entry



\*\*\*\*CONCENTRATIONS AND DOSAGES WITH 10 DEGREE UNCERTAINTIES\*\*\*\*

RANGE(M): 5000.0

AZIMUTH(DEG): 170.0

MATERIAL	CONCENTRATION (PPM)		DOSAGE (PPM)	
HCL	.184E+01 +/-	.050	109.108 +/-	57.684
CO	.676E-02 +/-	.000	7.730 +/-	.211
CO2	.403E+01 +/-	.110	4606.357 +/-	125.984
AL2O3	.473E+01 +/-	.129	5409.617 +/-	147.953
NO	.322E-02 +/-	.000	3.681 +/-	.101

RANGE(M): 5000.0

AZIMUTH(DEG): 190.0

MATERIAL	CONCENTRATION (PPM)		DOSAGE (PPM)	
HCL	.306E+01 +/-	.285	3494.906 +/-	326.191
CO	.112E-01 +/-	.001	12,809 +/-	1.195
CO2	.668E+01 +/-	.623	7632.983 +/-	712.411
AL2O3	.784E+01 +/-	.732	8964.029 +/-	836.641
NO	.534E-02 +/-	.000	6.099 +/-	.569

RANGE(M): 7500.0

AZIMUTH(DEG): 190.0

MATERIAL	CONCENTRATION (PPM)		DOSAGE (PPM)	
HCL	.212E+01 +/-	.219	2428.497 +/-	250.218
CO	.779E-02 +/-	.001	8.900 +/-	.917
CO2	.464E+01 +/-	.478	5303.912 +/-	546.484
AL2O3	.545E+01 +/-	.561	6228.813 +/-	641.781
NO	.371E-02 +/-	.000	4.238 +/-	.437

RANGE(M): 10000.0

AZIMUTH(DEG): 190.0

MATERIAL	CONCENTRATION (PPM)		DOSAGE (PPM)	
HCL	.163E+01 +/-	.176	1859.782 +/-	201.469
CO	.596E-02 +/-	.001	6.816 +/-	.738
CO2	.355E+01 +/-	.385	4061.821 +/-	440.015
AL2O3	.417E+01 +/-	.452	4770.125 +/-	516.745
NO	.284E-02 +/- +/-	.000	3.246 +/-	.352

RANGE(M): 12500.0

AZIMUTH(DEG): 190.0

MATERIAL	CONCENTRATION (PPM)		DOSAGE (PPM)	
HCL	.132E+01 +/-	.147	1506.710 +/-	168.214
CO	.483E-02 +/-	.001	5.522 +/-	.616
CO2	.288E+01 +/-	.321	3290.701 +/-	367.386
AL2O3	.338E+01 +/-	.377	3864.536 +/-	431.451
NO	.230E-02 +/- +/-	.000	2.630 +/-	.294

Figure 3-18. Off-Centerline Concentrations and Dosages with  $\pm 10^0$  Uncertainties for all Effluents



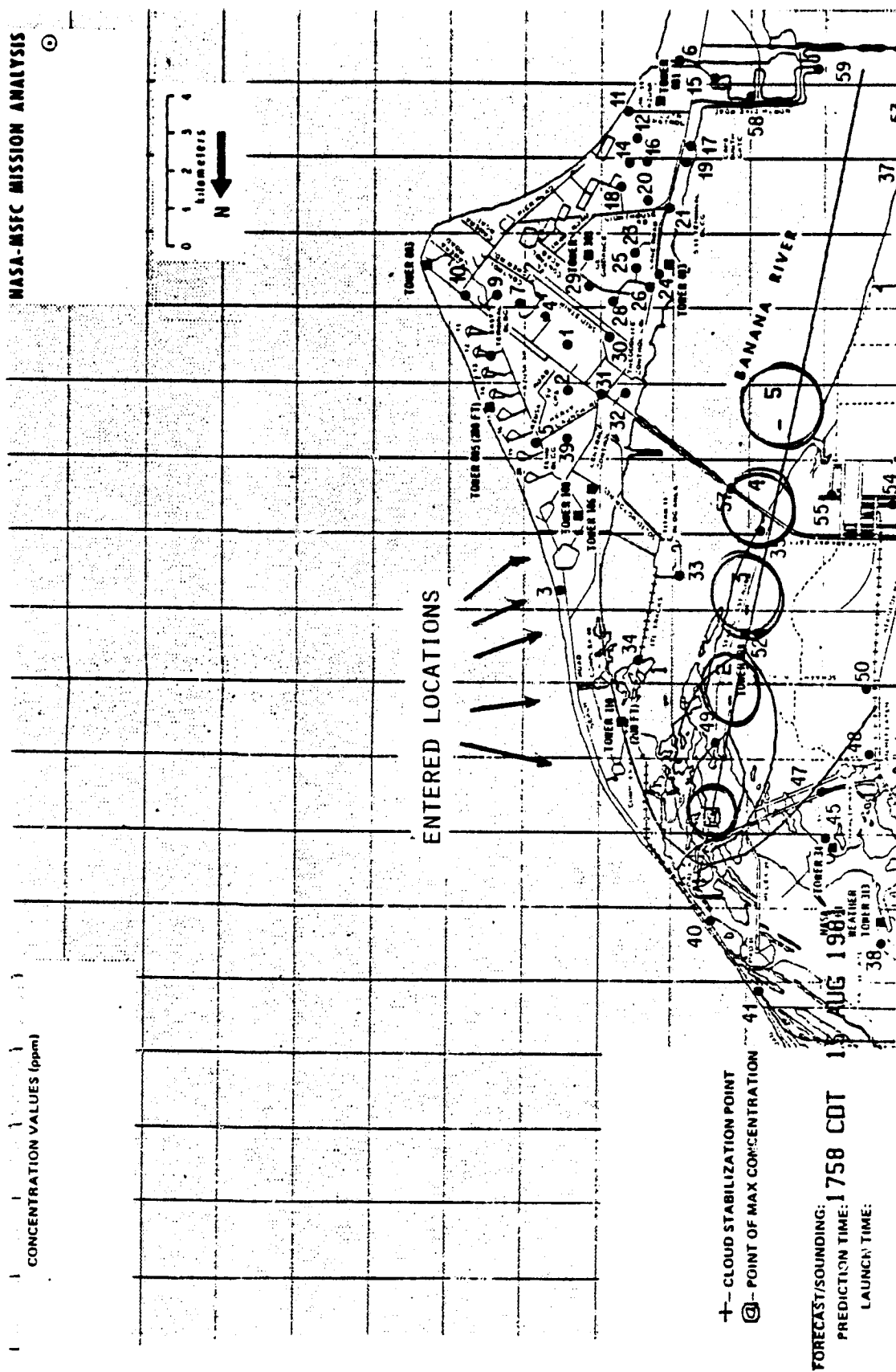


Figure 3-19. Off-Centerline Point Plotting



---

CONCENTRATION ISOPLETH PLOT DESIRED?

YES

NO

YES

DEFAULT ISOPLETH CONCENTRATION VALUES ARE: CONC(1) = .515  
CONC(2)=2.573      CONC(3)=3.859      CONC(4)=1.0

USE DEFAULT ISOPLETH CONCENTRATION VALUES ?

YES

NO

YES

T

\*\*\*\*\* PROGRAM REEDS HAS TERMINATED NORMALLY \*\*\*\*\*  
REE05 : STOP 0000

---

Figure 3-20. Default Isopleth Values, Isopleth Plotting,  
and Program Termination

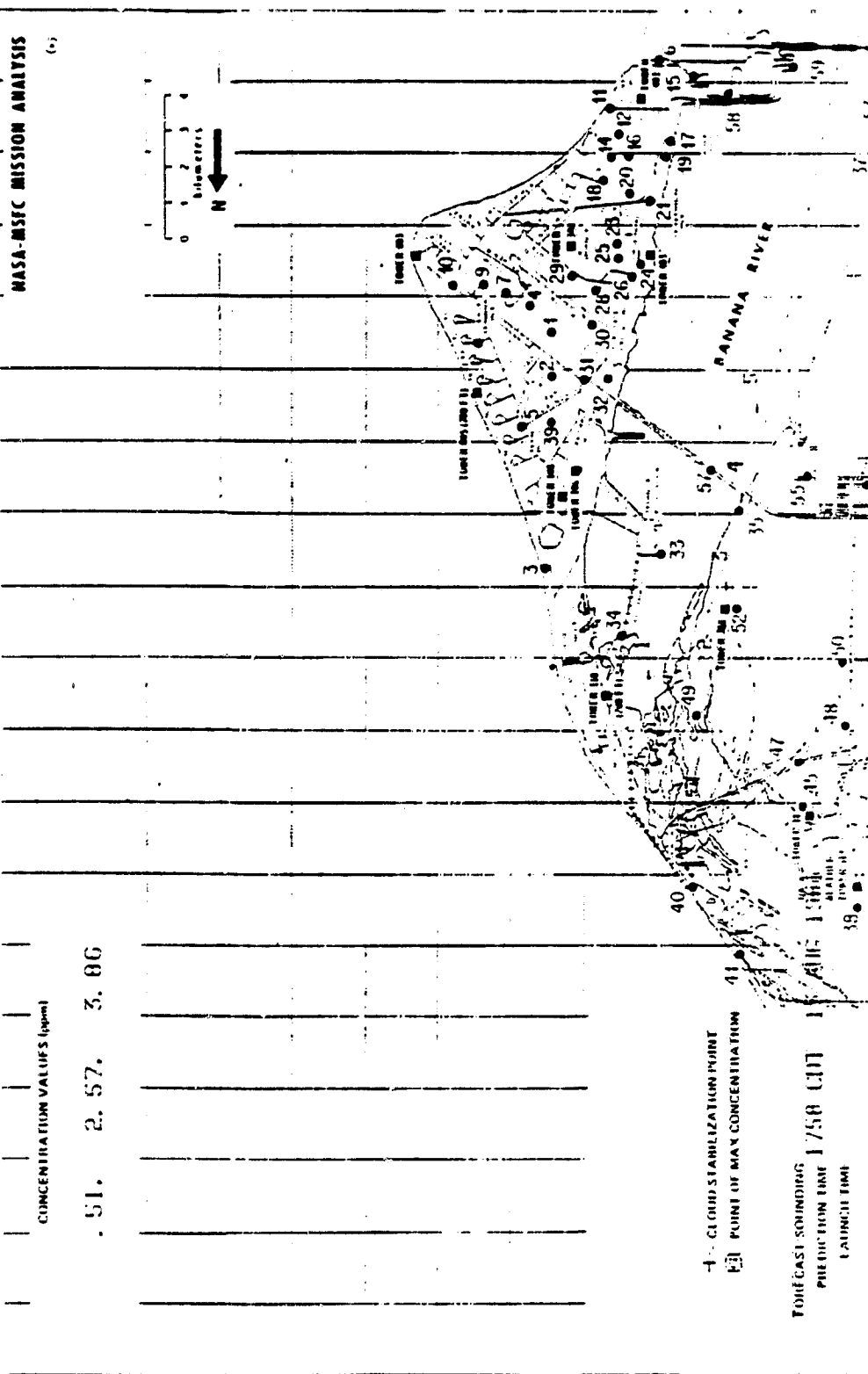


Figure 3-21. Isopleth Plotting



CLOUD LOCATION AND DIMENSIONS			
TIME FROM LAUNCH (MIN)	(SEC)	RANGE (METERS)	AZIMUTH (DEG)
		0.000	
THE DISTANCE FROM STABILIZATION IS:			
2	54.0	119.8	201.3
THE DISTANCE FROM STABILIZATION IS:			
7	54.0	304.4	180.9
THE DISTANCE FROM STABILIZATION IS:			
12	54.0	498.3	176.1
THE DISTANCE FROM STABILIZATION IS:			
17	54.0	693.8	174.0
THE DISTANCE FROM STABILIZATION IS:			
22	54.0	889.8	172.8
THE DISTANCE FROM STABILIZATION IS:			
27	54.0	1086.0	172.1
THE DISTANCE FROM STABILIZATION IS:			
32	54.0	1282.3	171.5
THE DISTANCE FROM STABILIZATION IS:			
37	54.0	1478.7	171.2
THE DISTANCE FROM STABILIZATION IS:			
42	54.0	1675.2	170.9
THE DISTANCE FROM STABILIZATION IS:			
47	54.0	1871.7	170.6
THE DISTANCE FROM STABILIZATION IS:			
52	54.0	2068.2	170.4
THE DISTANCE FROM STABILIZATION IS:			
57	54.0	2264.8	170.3
THE DISTANCE FROM STABILIZATION IS:			
62	54.0	2461.3	170.2

DIAMETERS (METERS)	
CROSS WIND	ALONG WIND
1285.3	1285.3
1515.0	1285.6
1756.7	1286.4
2006.1	1287.7
2260.8	1289.5
2519.0	1291.9
2779.8	1294.7
3042.6	1298.1
3306.9	1302.0
3572.3	1306.4
3838.6	1311.3
4105.6	1316.7
4373.2	1322.5

Figure 3-22. Cloud Position and Extent as a Function of Time





## SECTION 4.0 LISTINGS

This section contains the current listings of critical elements of the REEDS system. They are included as follows:

### SECTION 4.1 Main Program Modules and Subroutines Packages

- 4.1.1 REEDS (Executive Module)
- 4.1.2 REEDT
- 4.1.3 RCLDS
- 4.1.4 RCLDT
- 4.1.5 RMTPS
- 4.1.6 RMTPT
- 4.1.7 RCONS
- 4.1.8 RCONT
- 4.1.9 RISOS
- 4.1.10 RISOT

### SECTION 4.2 AUXILIARY SUBROUTINE LISTINGS

- 4.2.1 CTLB2
- 4.2.2 MLIB2
- 4.2.3 CALIX
- 4.2.4 DLINE

### SECTION 4.3 MAIN MODULE QUESTION FILES

- 4.3.1 ?REEDS
- 4.3.2 ?RCLDS
- 4.3.3 ?RCONS
- 4.3.4 ?RISOS

### SECTION 4.4 EXAMPLE METEOROLOGICAL DATA FILES



4.1 MAIN PROGRAM MODULES AND SUBROUTINE PACKAGES

This section contains the current listings of current Main Program Modules and the respective Main Subroutine Packages. They are listed according to the sequences indicated in Section 4.0.



ORIGINAL PAGE IS  
OF POOR QUALITY

```

&SPEED5 T=00003 IS ON CR00003 USING 00150 BLKS R=0000
0001 FTMX,L
0002 PROGRAM REEDS(3), REEDS SINGLE LAYER DIFFUSION MODEL--C48810825
0003 C
0004 C *****
0005 C *
0006 C * NASA/NSFC REED Code -- -- EXPERIMENTAL
0007 C *
0008 C * SINGLE LAYER DISTRIBUTION TECHNIQUE
0009 C *
0010 C * MAIN PROGRAM -- REEDS
0011 C *
0012 C *****
0013 C
0014 C *****
0015 C *
0016 C * LOADING INSTRUCTIONS:
0017 C *
0018 C * REEDS, RCLDS, RCONS, AND RISOS ARE LOADED AS FOLLOWS:
0019 C *
0020 C * LOAD %RXXXS
0021 C * MERGE %RXXXT
0022 C * MERGE %XLINE
0023 C * MERGE %DLINE
0024 C * MERGE %CALIX
0025 C * MERGE %CTLB2
0026 C *
0027 C *
0028 C *
0029 C *
0030 C *
0031 C *
0032 C *
0033 C *
0034 C *
0035 C *****
0036 C >>> COMMON AREAS
0037 C
0038 C
0039 C >>> MATH PARAMETERS
0040 C
0041 C COMMON PI, P10VR2, P143, TMOPT, SQR2PI, DELRAD, RADHEC, TV2
0042 C >>> VEHICLE PARAMETERS
0043 C
0044 C COMMON VPARK(10), SIGMCL
0045 C >>> OPTION PARAMETERS
0046 C
0047 C COMMON YPOS, YES, ISNGFG, IVHCL, IRUN, ICALE, IPLACE, ITIMEZ, IGOIMP,
0048 C MURRUH, LU, NUM, MOULY, MFORM, LUPPHT, T100, IFL1, IFL2, IFL3
0049 C INTEGER YES
0050 C >>> NET DATA
0051 C
0052 C COMMON ALT(30), IDIR(30), SPEED(30), TEMP(30), PFI(5), Z(30), PFI(10), Z(30),
0053 C SUPDEN, FILE(3)
0054 C INTEGER FILE
0055 C >>> CALCULATION TIME
0056 C
0057 C COMMON ITIME, TINY, TMOOTH(2), TVTAP
0058 C

```

```

0059 C >>> SOUNDING TIME
0060 COMMON 1STIME, ISDAY, ISMON(2), ISYEAR
0061 C
0062 C >>> LAUNCH TIME
0063 COMMON LTIME, LTIM, LDAY, LMONTH(2), LYEAR, LAUNTIME(10)
0064 LOGICAL LTIME, LVERSH
0065 C
0066 C >>> CLOUD STABILIZATION AND LAYER PARAMETERS
0067 COMMON STBLT, STBHT, STBZD, STDRNG, STBTIN, CLDRAD, RISTIN(30), BOTLAY,
0068 TOPLAY, CALHT, LAYTOP, LAYBOT, REFLEC, IPTZ
0069 C
0070 C >>> DIFFUSION COEFFICIENTS
0071 COMMON SIGX0, SIGX, SIGZ0, SIGAP, XDIST, YDIST, EXPZ, SOSIGZ, SIGY, SIGAL,
0072 CLOUDNG, AVSFD, AVDIR, SIGAZ
0073 C
0074 C >>> CONCENTRATION AND DOSAGE VALUES AND RANGES
0075 COMMON CONMAX, CONCPK, DOSMAX, DOSPK, RNGSNC, RINGSND
0076 C
0077 C >>> EQUIVALENCE STATEMENT FOR VEHICLE PARAMETERS
0078 EQUIVALENCE (QC1, VPAR(1)), (QC2, VPAR(2)), (QC3, VPAR(3)),
0079 (QT1, VPAR(4)), (QT2, VPAR(5)), (QT3, VPAR(6)),
0080 (QA, VPAR(7)), (QB, VPAR(8)), (CC, VPAR(9)),
0081 (HEATH, VPAR(10)), (HEATH, VPAR(11)), (HEATA, VPAR(12)),
0082 (PHCL, VPAR(13)), (PCO, VPAR(14)), (PCO2, VPAR(15)),
0083 (PAL203, VPAR(16)), (PHO, VPAR(17)), (GAMMAX, VPAR(18))
0084 DIMENSION ILINE(32), IDATA(10), IERS(32), JUTIN(5), IIMG(5.2)
0085 DIMENSION IDAYC(5), ITING(5)
0086 C
0087 C INPUT FORMAT STATEMENTS
0088 C
0089 90 FORMAT(A1)
0090 91 FORMAT(I2, I1, I2A2, I2)
0091 100 FORMAT(I2, I1, I2A2, I2)
0092 101 FORMAT(A1)
0093 102 FORMAT(I10A2)
0094 103 FORMAT(I4, I5I2, I1XA2, A1, I1X14)
0095 104 FORMAT(I4, I3I2, I1XA2, A1, I1X14)
0096 105 FORMAT(I6, I1X13, I1X4, I1F6, I1F7.2, I1X7.2)
0097 106 FORMAT(I4, " C" A2, I1X2, I1, A2, A1, I1, I14)
0098 107 FORMAT(I2, I1, A2, A1, I1, I14)
0099 108 FORMAT(F7.2)
0100 109 FORMAT(F4.1)
0101 110 FORMAT(F4.1)
0102 111 FORMAT(I2, I1, A2, A1, I1, I14)
0103 C
0104 C OUTPUT FORMAT STATEMENTS
0105 C
0106 200 FORMAT ("100(IH+)/IX, 80(IH+)/IX, 100(IH+)/50X, 100(IH+)/
0107 IX, 10(IH+)/" MMSA, MSFC REED CODE USING SINGLE LAYER "
0108 "MODEL" 3X "21 DEC 1979 " , 10(IH+)/IX, 10(IH+)/, 60X, 10(IH+)/
0109 IX, 10(IH+)/, 60X, 10(IH+)/IX, 10(IH+)/, 27X " REEDS " 27X, 10(IH+)/
0110 IX, 10(IH+)/, 60X, 10(IH+)/IX, 30(IH+)/IX, 00(IH+)/
0111 "OPUH " 12 " USING DATA FILE " 302/
0112 "0" 3A2, A1 " LAUNCH VEHICLE "
0113 201 FORMAT ("0 LAUNCH TIME: " I4, I1X1, A2, 4X "DATE: " I2, I1X2, A1, I1X14)
0114 202 FORMAT ("0 PEPERATION TIME: " I4 " C" A2, 4X "DATE: " I2, I1X2, A1, I1X14)
0115 "0 DATA FILE HEADER INFORMATION: ")
0116 203 FORMAT ("0<<<<<<OPEN ERROR " I4 " , PROCESSING CONTINUES WITH "
0117 "NEXT PUP)>>>")
0118 204 FORMAT ("0<<<<<<PAGE ERRORS " I4 " , PROCESSING CONTINUES WITH "

```

ORIGINAL PAGE IS  
OF POOR QUALITY

```

NEXT RUN>>>
205 FORMAT (6X,40A2)
206 FORMAT ('05X"TIME: "I4,IXA1,A2,4X"DATE: "I2,IXA2,A1,IXI4)
207 FORMAT ('/'0"60(IHS)/IX,20(IHS),40X,20(IHS)/
IX,20(IHS),16X"SHOOTING"16X,20(IHS)/
IX,20(IHS),40X,20(IHS)/IX,20(IHS)/IX,80(IHS)/
208 FORMAT ('/'0"60(IHF)/IX,20(IHF),40X,20(IHF)/
IX,20(IHF),16X"FORECAST"16X,20(IHF)/
IX,20(IHF),40X,20(IHF)/IX,80(IHF)/
209 FORMAT ('0SURFACE DENSITY (GM/M**3): "F8.2)
210 FORMAT ('0LAYER ALTITUDE DIRECTION SPEED TEMP
"POI-TEMP D P TEMP PRESSURE"/
"NO. (FEET) (METERS) (DEGREES) (M/SEC)
" (DEGREES CENTIGRADE) (MILLIBARS)")
211 FORMAT (2X12,17,2X15,7X13,5XF4.1,4XF4.1,4XFS.2,6XF4.1,6XFS.2)
212 FORMAT ('***OPEN OR IDRPL IERR: "I3", JOB TERMINATED***')
213 FORMAT ('***IDRPL IERR: "I3", JOB TERMINATED***')

TYPE AND DIMENSION STATEMENTS
INTEGER BLANKS,RCHAR,TCHAR,SCNAR,VNAMES(4,3),RUMIUM,RA,
FO,SDY,TE,ZERO0,ZEPOI,RCLDR(3),NAMRP(3,5)
DIMENSION IPARC(5),VPARS(10,5),IDCB(272),IBUF(40),IALI(30),
DPTMP(30),NAME(3),NAMEF(3)

DATA STATEMENTS
DATA NAME/2H=R,2HEE,2HD2/
DATA IERS/32*H /
DATA NAMEF/2H7R,2HEE,2HDS/

*****
VPARS(1 THRU 10) = SHUTTLE (19 - 36) = TITAN (37 - 54)=DELTA
<55 - 72) = DELTA 3914 (73 - 90) = MINUTEMAN
*****

VALUES CONSISTENT WITH SAI REVISION OF SS EIS---2/2278
Q01,Q11 ARE VALUES FOR NORMAL LAUNCH SEQUENCE
Q02,Q12 ARE VALUES FOR ONE SRM BURNING ON LAUNCH PAD
Q03,Q13 ARE VALUES FOR BOTH SRMS EXPLODING ON LAUNCH PAD
VALUES ARE CONSISTENT WITH THIOKOL VAB SAFETY STUDY
BURN TIME IS 7.7778 TIMES NORMAL BURN TIME
CC IS HOLD DOWN TIME ON LAUNCH PAD FOR NORMAL LAUNCH
DATA VPARS/9.478590903E6,3.84505682E6,9.88726071E5,1.251174E9,
5.075475E8,1.015025E9,.652213,.468085,
.29653,1479.7,1062.35,1000....1146,.00042,.25029,18279,
.37500,1479.7,1062.35,1000....1146,.00042,.25029,18279,
.00019,.64,
.00020,.64,
DATA VPARS/1.5454E7,6.882969E6,9.88726071E5,1.834794173E9,
DATA VPARS/1.5219E7,6.882968E6,9.88726071E5,1.251174E9,
6.563295E9,1.713859E9,.652213,.468085,
5.075475E8,1.015025E9,.652213,.468085,
.29653,1479.7,1062.35,1000....1146,.00042,.25029,18279,
.37500,1479.7,1062.35,1000....1146,.00042,.25029,18279,
.00019,.64,
.00020,.64,
5.437528E6,2.718744E6,1.350302E6,3.267518E0,
1.6312594E9,3.2629164E9,4.935947E9,5.191107,
5.0,2.021,1,1010.57,1000.0,1032.005,0.029,
5.0,2.021,1,1010.55,1000.0,1032.005,0.029,
5.025 0005 74

```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0179      8.360685E5,9.09611E4,2.729434E5,2.887538E7,  
0180      3.14229E6,1.88537E7,922156,432703,54,1766,0,  
0181      1000,0,690,0,1866,2055,0156,3391,0002,  
0182      1000,0,690,0,1210,0156,2055,2214,0002,  
0183      .50,  
0184      1.057557E6,1.482923E5,3.70731E5,6.70269E7,  
0185      9.396616E6,4.699308E7,1.245756,4180947,  
0186      0,0,1449,9,1000,0,411,10,1866,2055,0156,3391,  
0187      0,0,1449,9,1000,0,411,10,1589,0331,2783,1936,  
0188      .0002,50,  
0189      4.684476E5,4.604476E5,1.171119E5,2.8186856E7,  
0190      2.8186856E7,2.8186856E7,469982,463333,0,0,  
0191      2055,9,2055,9,1000,0,1866,2055,0156,3391,  
0192      2055,9,2055,9,1000,0,1866,0156,2055,3391,  
0193      .0002,647  
0194 C  
0195 DATA BLANKS/2H /, RCHAR/1HR/, TCHAR/1HT/, SCHAR/1HS/,  
0196 RA/2HRA/, FO/2HFO/, SDT/2HDT/, TE/2HTE/, ZER00,3006600/,  
0197 ZER01/0344718/, ACLDR/2HRC,2HLD,1HS/  
0198 DATA VNAME$;2HSH,2HUT,2HTL,1HE,  
0199      2HTI,2HTA,1HR,1H /  
0200      2HDE,2HLT,1HA,1H /  
0201 DATA NAMRP/2HRC,2HLD,2HS ,2HRN,2HTP,2HS ,2HRC,2HOM,2HS ,  
0202      2HRI,2HSO,2HS ,2HRH,2HTO,2HS /  
0203 C  
0204 C  
0205 C  
0206 C  
0207 C  
0208 C  
0209 C  
0210 C  
0211 C  
0212 C  
0213 C  
0214 C  
0215 C  
0216 C  
0217 C  
0218 C  
0219 C  
0220 C  
0221 C  
0222 C  
0223 C  
0224 C  
0225 C  
0226 C  
0227 C  
0228 C  
0229 C  
0230 C  
0231 C  
0232 C  
0233 C  
0234 C  
0235 C  
0236 C  
0237 C  
0238 C
```

SET THE LOGICAL UNIT NUMBERS OF THE DEVICES TO BE USED FOR  
INPUT AND PRINT

```
CALL RMPAR(IPAR)  
LU = MAX0(IPAR(1),1)  
LUPRINT = IPAR(2)  
IF(LUPRINT.LE.0) LUPRINT = 6  
IPL1 = IPAR(3)  
IF(IPL1.LE.0) IPL1 = 12  
IPL2 = IPAR(4)  
IF(IPL2.LE.0) IPL2 = 20  
IPL3 = IPAR(5)  
IF(IPL3.LE.0) IPL3 = 21
```

FIND OUT IF THIS IS THE CRT OR FLASMASCOPE VERSION OF REEDS  
BEING RUN

```
CALL VERSKT)  
LVERSH = 1 .EQ. 0
```

IF ALL SEGMENTS --- IF ERRORS ENCOUNTERED, WRITE OUT AN ERROR  
MESSAGE AND TERMINATE REEDS

```
NUMSEG = 4  
IF(LVERSH) NUMSEG = 5  
DO I = 1, NUMSEG  
CALL OPEN(ICDB,IERR,NAMPP(1),1)  
CALL IDRFL(ICDB,IERR,NAMPP(1),1)  
IF(IERR.NE.0) GO TO 1  
WRITE (LU,212) IERR  
STOP  
1 CONTINUE
```

FOR PLASMASCOPE ONLY CALL READ TO INITIALIZE REEDS

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0239 C          GRAF FOR (RT)
0240 C
0241 C          CALL GRAF(1)
0242 C
0243 C          BEGIN PROCESSING OF NEW DATA BY CLEARING SCORE
0244 C
0245 C          2 CALL CLEAR
0246 C
0247 C          INITIALIZE SOME COMMON VARIABLES
0248 C
0249 C          LTIME = .FALSE.
0250 C          YES = IHY
0251 C          NO = IHN
0252 C          PI = 3.141593
0253 C          P1OVR2 = 0.5 * PI
0254 C          P143 = 1.3333333 * PI
0255 C          TMOPI = 2.0 * PI
0256 C          SOR2PI = SORT(TMOPI)
0257 C          DEGRAD = PI/180.0
0258 C          RADDEC = 180.0/PI
0259 C          ISHDFD = 0
0260 C          IVMICL = 0
0261 C          IRUN = 0
0262 C          ICALC = 0
0263 C          IFLADE = 0
0264 C          ITIMEZ = 0
0265 C          IGOIMP = 0
0266 C          LAYBOT = 0
0267 C          BOTLAY = 0.0
0268 C          CALMT = 0.0
0269 C          REFLEC = 1.0
0270 C          REFLEC = 0.1
0271 C          IPTZ = 0
0272 C
0273 C          WRITE THE HEADER ON THE CONSOLE
0274 C
0275 C          CALL CLEAR
0276 C          YPOS = 490.0
0277 C          CALL DREAD(NAMEF,2,ILINE)
0278 C          CALL LERS(YPOS,LVERSH)
0279 C          CALL CHAR(0.0,YPOS,0,ILINE,64,0.0)
0280 C          CALL GETD(LTIM,LDAY,LMONTH,LYEAR)
0281 C          CALL CODE
0282 C          WRITE (IDATAF,107) LDAY,LMONTH,LYEAR
0283 C          CALL CHAR(368.0,YPOS,0,IDATAF,11,2.0)
0284 C          YPOS = YPOS - 32.0
0285 C
0286 C          CBJ*** DECIDE ON THE TYPE OF CLOUD RISE DESIRED...NEW CURVE FIT OR
0287 C          CBJ*** OLD PROCEDURE
0288 C          WRITE(1,777)
0289 C          FORMAT(" DO YOU WANT THE LOW ALT CURVE FIT? YES NO ")
0290 C          READ(1,776) IANS
0291 C          FORMAT(" ",")
0292 C          IPTZ = I
0293 C          IX = 25
0294 C          IF(IANS .NE. IHY) GO TO 771
0295 C          IX = 30
0296 C          IPTZ = 0
0297 C          WRITE(1,779)
0298 C          FORMAT(" LOW ALT CURVE FIT? YES NO ")
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0299 GO TO 770
0300 CONTINUE
0301 WRITE(1,773)
0302 FORMAT(" LOW ALTITUDE CURVE F37".J0X," NO")
0303 770
0304 CONTINUE
0305 YPOS = YPOS - 32.0
0306 READ IN THE 1ST 4 CHAR'S OF THE DATA FILE NAMES
0307 C
0308 CALL DREAD(NAMEF,3,ILINE)
0309 CALL LERS(YPOS,LVERSH)
0310 CALL CHAR(0.0,YPOS,0,ILINE,43,3,0)
0311 CALL CHAR(384.0,YPOS,0,ILINE(25),8,3,0)
0312 CALL CHAR(464.0,YPOS,0,ILINE(30),6,0,0)
0313 C
0314 C IF PLASMASCOPE VERSION, CALL "IN".
0315 C
0316 IF(LVERSH)GO TO 3
0317 C
0318 C OTHERWISE RESPOND WITH Y FOR YES, N FOR NO TO ANSWER QUESTIONS
0319 C*** A "0" FOR ALMOST ANY ENTRY WILL PERMIT IMMEDIATE TERMINATION
0320 C*** OF THE PROGRAM
0321 C***
0322 READ(LU,90) IANS
0323 IF(IANS.EQ.1)GO TO 91
0324 C
0325 C "YES" IS THE DEFAULT CASE
0326 C
0327 IX = 25
0328 IF(IANS .NE. NO) GO TO 4
0329 C
0330 C THIS IS THE "NO" RESPONSE---ENTER THE # OF RUNS, AND THE
0331 C COMMON DATA FILE NAME
0332 C
0333 IX = 30
0334 GO TO 4
0335 3 CALL IK(1,JTYPE,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0)
0336 4 CALL CHAR(0.0,YPOS,0,ILINE,64,0,0)
0337 IF(IX .LE. 25)CALL CHAR(464.0,YPOS,0,IERS,6,0,0)
0338 IF(IX .GT. 25)CALL CHAR(384.0,YPOS,0,IERS,8,0,0)
0339 YPOS = YPOS - 32.0
0340 IF(IX .LE. 25)GO TO 6
0341 CALL DREAD(NAMEF,4,ILINE)
0342 CALL LERS(YPOS,LVERSH)
0343 CALL CHAR(0.0,YPOS,0,ILINE,64,0,0)
0344 YPOS = YPOS - 16.0
0345 CALL DREAD(NAMEF,5,ILINE)
0346 CALL LERS(YPOS,LVERSH)
0347 CALL CHAR(0.0,YPOS,0,ILINE,22,3,0)
0348 MIN = 9
0349 CALL BLANK(IDATAF,10)
0350 CALL IK(0,JTYPE,175.0,YPOS,0,DATAF,1111,0.31,0.31,13,13,13)
0351 CALL CHAR(0.0,YPOS,0,ILINE,22,0,0)
0352 YPOS = YPOS - 32.0
0353 CALL CODE(MIN)
0354 READ (IDATAF,100) INPRIN,FILE(1),FILE(2), IFOFF
0355 INPRIN = MAX(INPRIN,1)
0356 IF( IFOFF .GT. 0) IFOFF = IFOFF - 1
0357 IF( FILE(1) .NE. BLANK)GO TO 7
```



```

0359 FILE(2) = 2HTA
0360 I-OFF = 0
0361 NUMRUN = 1
0362 7 IFOFF1 = IFOFF + 1
0363 IPLACE = 0
0364 ISNDG = 9
0365 CBJ
0366 CBJ*** DECODE LOCATION FROM DATA FILE NAME
0367 CBJ
0368 IF(FILE(1).EQ.2HTA .AND. FILE(2).EQ.2HND)IPLACE = 1
0369 IF(FILE(1).EQ.2HTA .AND. FILE(2).EQ.2HPE)IPLACE = 2
0370 IF(FILE(1).EQ.2HTA .AND. FILE(2).EQ.2HTA)IPLACE = 3
0371 IF(FILE(1).EQ.2HSP .AND. FILE(2).EQ.2HEC)IPLACE = 4
0372 CBJ
0373 CBJ*** ACCORDING TO PLACE, TIME ZONE CODE WILL BE ENCODED
0374 ITIMEZ = IHE
0375 IF(IPLACE .EQ. 1)ITIMEZ = IMP
0376 IF(IPLACE .EQ. 2)REWIND 8
0377 IF(IPLACE.EQ.4) ISNDG = 1
0378 IF(IPLACE.NE.4)GO TO 9
0379 C
0380 CBJ*** IF THIS IS A SPECIFIC SOUNDING RUN, SOUNDING CODE WILL BE SET=1
0381 C
0382 C** ENTER SPECIFIC SOUNDING TIME AND DATE
0383 C
0384 IF(NUMRUN.GT.5) NUMRUN = 5
0385 CALL DREAD(NAMEF,26,ILINE)
0386 CALL LERS(YPOS,LVERSH)
0387 CALL CHAR(0.,YPOS,0,ILINE,46,3,0)
0388 DO 8 NSPEC=1,NUMRUN
0389 YPOS = YPOS - 16.
0390 MRUN = 26 + NSPEC
0391 CALL DREAD(NAMEF,MRUN,ILINE)
0392 CALL LERS(YPOS,LVERSH)
0393 CALL CHAR(0.,YPOS,0,ILINE,20,0,0)
0394 CALL BLANK(IDATAF,10)
0395 IIN = 12
0396 CALL IH(0,JTYPE,160.,YPOS,0,IDATAF,IIN,0,31,0,31,IX,IY)
0397 CALL CODE(IIN)
0398 8 READ(IDATAF,111) IDAY(NSPEC),IHONG(NSPEC,1),IHONG(NSPEC,2),
0399 ITIME(NSPEC)
0400 YNPOS = NUMRUN*16. + YPOS
0401 CALL DREAD(NAMEF,26,ILINE)
0402 CALL CHAR(0.,YNPOS,0,ILINE,46,0,0)
0403 YPOS = YPOS - 32.
0404 C
0405 C FIND OUT IF THESE RUNS ARE TO BE RESEARCH RUNS < INTERACTION
0406 C AND PLOTTING ALLOWED > OR PRODUCTION RUNS
0407 C
0408 9 CALL DREAD(NAMEF,7,ILINE)
0409 CALL LERS(YPOS,LVERSH)
0410 CALL CHAR(0,0,YPOS,0,ILINE,12,3,0)
0411 CALL CHAR(96,0,YPOS,0,ILINE(7),19,0,0)
0412 CALL CHAR(240,0,YPOS,0,ILINE(16),14,3,0)
0413 CALL CHAR(352,0,YPOS,0,ILINE(23),18,0,0)
0414 C
0415 C*** CHECK FOR PLASMASCOPE.....
0416 IF(LVERSH.GO TO 10
0417 C
0418 C*** IF CBT SET BY ACCORDING TO B-ELECTRONICALLY CONTROLLED

```

ORIGINAL PAGE IS  
OF POOR QUALITY

```

0419 C*** OR P (PRODUCTION)
0420 C
0421 IX = 15
0422 READ(LU,90) IANS
0423 IF(IANS.EQ.10)GO TO 81
0424 IF( IANS.NE.10) .AND. ( IANS.NE.10) ) GO TO 11
0425 IX = 10
0426 IF(IANS .NE. 10)GO TO 11
0427 IX = 20
0428 IF(IANS .EQ. 10)GO TO 11
0429 CONTINUE
0430 CALL IN(1,JTYPE,0.0,0.0,0.0,0.31,0.31,IX,IY)
0431 CONTINUE
0432 CALL CHAR(0.0,YPOS,0,ILINE,64,0,0)
0433 IF(IX.LT.12)CALL CHAR(240.0,YPOS,0, IERS,32,0,0)
0434 IF(IX.LT.12)IRUN = 1
0435 IF(IX.GE.12 .AND. IX.LT.19)CALL CHAR(120.0,YPOS,0, IERS,16,0,0)
0436 IF(IX.GE.12 .AND. IX.LT.19)IRUN = 2
0437 IF(IX.GE.12 .AND. IX.LT.19)CALL CHAR(368.0,YPOS,0, IERS,16,0,0)
0438 IF(IX.GE.19)CALL CHAR(120.0,YPOS,0, IERS,30,0,0)
0439 IF(IX.GE.19) IRUN = 3
0440 YPOS = YPOS - 32.0
0441 IF(IRUN .LE. 2)GO TO 12
0442 C
0443 C FOR PRODUCTION RUNS, READ IN THE TOP OF THE SURFACE LAYER
0444 C AND THE SIGMA OF THE WIND AZIMUTH ANGLE TO BE USED FOR ALL RUNS
0445 C
0446 CALL BREAD(NAMEF,11,ILINE)
0447 CALL LERS(YPOS,LVERSH)
0448 CALL CHAR(0.0,YPOS,0, ILINE,33,3,0)
0449 NIN = 6
0450 CALL BLANK(IDATAF,10)
0451 CALL IN(0,JTYPE,263.0,YPOS,0, IDATAF.NIN,0.31,0.31,IX,IY)
0452 CALL CHAR(0.0,YPOS,0, ILINE,33,0,0)
0453 CALL CODE(NIN)
0454 READ (IDATAF,108) TOPLAY
0455 YPOS = YPOS - 32.0
0456 CALL BREAD(NAMEF,12,ILINE)
0457 CALL LERS(YPOS,LVERSH)
0458 CALL CHAR(0.0,YPOS,0, ILINE,43,3,0)
0459 NIN = 2
0460 CALL BLANK(IDATAF,10)
0461 CALL IN(0,JTYPE,295.0,YPOS,0, IDATAF.NIN,0.31,0.31,IX,IY)
0462 CALL CHAR(0.0,YPOS,0, ILINE,37,0,0)
0463 CALL CODE(NIN)
0464 READ (IDATAF,109) ISIGAZ
0465 IF(NIN .EQ. 1)ISIGAZ = ISIGAZ/10
0466 SIGAZ = FLOAT(ISIGAZ)
0467 YPOS = YPOS - 32.0
0468 C
0469 C NET PLOTS FORMS? .....USE ONLY IF ABSOLUTELY NECESSARY
0470 C
0471 C CALL BREAD(NAMEF,24,ILINE)
0472 CALL LERS(YPOS,LVERSH)
0473 CALL CHAR(0.0,YPOS,0, ILINE,30,3,0)
0474 CALL CHAR(240.0,YPOS,0, ILINE(16),10,0,0)
0475 CALL CHAR(472.0,YPOS,0, ILINE(30),4,3,0)
0476 C
0477 C CALL BREAD(NAMEF,VERSION), CALL "III"

```





ON THIS PAGE IS  
OF POOR QUALITY

```
0593 C
0600 C THIS IS THE "NO" RESPONSE
0601 C
0602 IF<IANS.NE.NO> GO TO 22
0603 IX = 30
0604 GO TO 22
0605 CONTINUE
0606 CALL IHC<1, JTYPE, 0, 0, 0, 0, 0, 0, 0, 31, 0, 31, IX, IY>
0607 CONTINUE
0608 CALL CHAR<0, 0, YPOS, 0, ILINE, 20, 0, 0>
0609 CALL CHAR<384, 0, YPOS, 0, ILINE<25>, 15, 0, 0>
0610 IF<IX .LE. 25>CALL CHAR<464, 0, YPOS, 0, IERS, 6, 0, 0>
0611 IF<IX .GT. 25>CALL CHAR<384, 0, YPOS, 0, IERS, 8, 0, 0>
0612 YPOS = YPOS - 32.0
0613 IF<IX .LE. 25>GO TO 20
0614 CALL DREAD<NAMEF, 9, ILINE>
0615 CALL LERS<YPOS, LVERSH>
0616 CALL CHAR<0, 0, YPOS, 0, ILINE, 26, 3, 0>
0617 NIN = 20
0618 CALL BLANK<IDATAF, 10>
0619 CALL IHC<0, JTYPE, 267, 0, YPOS, 0, IDATAF, NIN, 0, 31, 0, 31, IX, IY>
0620 CALL CHAR<0, 0, YPOS, 0, ILINE, 26, 0, 0>
0621 CALL CODE<NIN>
0622 READ<IDATAF, 102> <LAUNTD<1>, I-1, 10>
0623 YPOS = YPOS - 32.0
0624 IF<LAUNTD<1>, .EQ. BLANKS>GO TO 20
0625 LTIME = .TRUE.
0626 CALL CODE<NIN>
0627 READ<LAUNTD, 103> LTIM, LDAY, LMONTH, LYEAR
0628 GO TO 21
0629 20 LAUNTD<4> = SDI
0630 C
0631 C READ IN THE LAUNCH VEHICLE, LET IT DEFAULT IF NOT ENTERED,
0632 C WRITE IT BACK OUT, AND FILL THE VPAR ARRAY WITH THE
0633 C APPROPRIATE VEHICLE PARAMETERS
0634 C
0635 21 CALL DREAD<NAMEF, 10, ILINE>
0636 CALL LERS<YPOS, LVERSH>
0637 CALL CHAR<0, 0, YPOS, 0, ILINE, 24, 3, 0>
0638 CALL CHAR<192, 0, YPOS, 0, ILINE<13>, 24, 0, 0>
0639 CALL CHAR<416, 0, YPOS, 0, ILINE<27>, 11, 3, 0>
0640 C
0641 C*** IF PLASMASCOPE, SKIP TO THE CALL TO "IN"
0642 IF<LVERSH>GO TO 217
0643 C
0644 C*** OTHERWISE RESPOND WITH D FOR DELTA, T FOR TITAN, OR S FOR SHUTTLE
0645 READ<LU, 90> IANS
0646 IF<IANS.EQ. IHO>GO TO 01
0647 C
0648 C*** LET A 0 ANSWER (BLANK) DEFAULT TO THE TITAN
0649 IF<IANS.NE.IHO> .AND. <IANS.NE.IHS> ) IX = 20
0650 IF<IANS.NE.IHO> .AND. <IANS.NE.IHT> ) IX = 30
0651 IF<IANS.EQ.IHO> IX = 10
0652 IF<IANS.EQ.IHS> IX = 30
0653 IF<IANS.EQ.IHT> IX = 20
0654 GO TO 218
0655 217 CONTINUE
0656 CALL IHC<0, JTYPE, 0, 0, 0, 0, 0, 0, 0, 31, 0, 31, IX, IY>
0657 218 CONTINUE
0658 CALL CHAR<0, 0, YPOS, 0, ILINE, 24, 0, 0>
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0659 C IF<IX .LE. 15>CALL CHAR<312,0,YPOS,0, IERS,24,0,0>  
0660 C IF<IX.GT.15 .AND. IX.LT.24>CALL CHAR<192,0,YPOS,0, IERS,12,0,0>  
0661 C IF<IX.GT.15 .AND. IX.LT.24>CALL CHAR<416,0,YPOS,0, IERS,11,0,0>  
0662 C IF<IX .GE. 24>CALL CHAR<192,0,YPOS,0, IERS,24,0,0>  
0663 C YPOS = YPOS - 32.0  
0664 C IF<IX .LE. 15>GO TO 25  
0665 C IF<IX .LE. 23>GO TO 24  
0666 C  
0667 C  
0668 C IF IVHCL = 0 THEN IT IS A SHUTTLE LAUNCH.  
0669 C  
0670 C IVHCL = 0  
0671 C GO TO 26  
0672 C  
0673 C  
0674 C  
0675 C IF IVHCL = 1 THEN IT IS A TITAN LAUNCH.  
0676 C  
0677 C  
0678 C 24 IVHCL = 1  
0679 C GO TO 26  
0680 C  
0681 C  
0682 C IF IVHCL = 2 THEN IT IS A DELTA LAUNCH.  
0683 C  
0684 C  
0685 C 25 IVHCL = 2  
0686 C 26 I = IVHCL + 1  
0687 C  
0688 C FILL THE VPAR ARRAY  
0689 C  
0690 C  
0691 C DO 28 J=1,16  
0692 C 28 VPAR(J) = VPAR(J,I)  
0693 C  
0694 C CHANGE IN BOTTOM LAYER WITH TOTAL REFLECTION  
0695 C (COMMENTED-OUT)  
0696 C ***THE NEXT FEW PAGES OF CODE WERE ORIGINALLY COMMENTED OUT AT THE  
0697 C ***DIRECTION OF DR. STEPHENS..BUT ARE TEMPORARILLY RESTORED..CABR10825  
0698 C  
0699 C CALL DREAD<NAMEF,15,ILINE>  
0700 C CALL LERS<YPOS,LVERSN>  
0701 C CALL CHAR<0,0,YPOS,0,ILINE,64,0,0>  
0702 C YPOS = YPOS - 16.0  
0703 C CALL DREAD<NAMEF,16,ILINE>  
0704 C CALL LERS<YPOS,LVERSN>  
0705 C CALL CHAR<32,0,YPOS,0,ILINE<3>,11,3,0>  
0706 C CALL CHAR<160,0,YPOS,0,ILINE<11>,43,6,0>  
0707 C IF<LVERSN> GO TO 291  
0708 C READ<LU,91> IANS  
0709 C IF<IANS.EQ.1N0>GO TO 81  
0710 C IX = 5  
0711 C IF<IANS .EQ. 2N1> IX = 12  
0712 C IF<IANS .EQ. 2N5> IX = 25  
0713 C GO TO 293  
0714 C 291 CONTINUE  
0715 C CALL INC<1,JTYPE,0,0,0,0,0,0,0,0,0,0,31,0,31,IX,IX>  
0716 C 293 CONTINUE  
0717 C CALL CHAR<0,0,YPOS,0, IERS,64,0,0>  
YF = YF05  
PAGE 6
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0719 CBJ CHECK FOR SURFACE -- STABILIZATION -- SOMETHING ELSE
0720 CBJ
0721 CBJ
0722 IF<IX .GE. 8)GO TO 29
0723 ICALC = 0
0724 CALL CHAR<0,0,YP,0,ILINE,16,0,0)
0725 GO TO 30
0726 IF<IX .GT. 20)GO TO 30
0727 ICALC = 1
0728 CALL CHAR<160,0,YP,0,ILINE(11),16,0,0)
0729 LAYBOT = 2
0730 BOTLAY = ALT(LAYBOT)
0731 CALJIT = BOTLAY
0732 GO TO 30
0733 CBJ
0734 CBJ
0735 CBJ
0736 30 ICALC = 2
0737 CALL CHAR<320,0,YP,0,ILINE(20),16,0,0)
0738 CALL DREAD<NAMEF,17,ILINE)
0739 CALL LERS<YPOS,LVERSH)
0740 CALL CHAR<0,0,YPOS,0,ILINE,42,3,0)
0741 CALL CHAR<384,0,YPOS,0,ILINE(25),0,3,0)
0742 CALL CHAR<464,0,YPOS,0,ILINE(30),6,0,0)
0743 CALL IH<1,JTYPE,0,0,0,0,0,31,0,31,IX,IY)
0744 CALL CHAR<0,0,YPOS,0,IERS,42,0,0)
0745 CALL CHAR<0,0,YPOS,0,ILINE,42,0,0)
0746 IF<IX .LE. 25)GO TO 37
0747 CALL CHAR<384,0,YPOS,0,IERS,8,0,0)
0748 YPOS = YPOS - 32.0
0749 CBJ
0750 CBJ
0751 CBJ
0752 ENTER HEIGHT OF CALCULATION, Zz
0753 CALL DREAD<NAMEF,18,ILINE)
0754 CALL LERS<YPOS,LVERSH)
0755 CALL CHAR<47,0,YPOS,0,ILINE(4),16,3,0)
0756 NIN = 6
0757 CALL BLANK<IDATF,10)
0758 CALL IH<0,JTYPE,128,0,YPOS,0,JDATF,NIN,0,31,0,31,IX,IY)
0759 PEAD<IDATF,*) CALHT
0760 CALL CHAR<0,0,YPOS,0,IERS,16,0,0)
0761 CALL CHAR<47,0,YPOS,0,ILINE(4),16,0,0)
0762 YPOS = YPOS - 32.0
0763 CBJ
0764 CBJ
0765 CBJ
0766 ENTER SURFACE REFLECTION?
0767 CALL DREAD<NAMEF,19,ILINE)
0768 CALL LERS<YPOS,LVERSH)
0769 CALL CHAR<0,0,YPOS,0,ILINE,45,3,0)
0770 CALL CHAR<383,0,YPOS,0,ILINE(25),0,3,0)
0771 CALL CHAR<472,0,YPOS,0,ILINE(30),6,0,0)
0772 CALL IH<1,JTYPE,0,0,0,0,0,31,0,31,IX,IY)
0773 CALL CHAR<0,0,YPOS,0,ILINE,64,0,0)
0774 IF<IX .LE. 25)PEFLEC = 1.0
0775 IF<IX .LE. 25)CNIL CHAR<464,0,YPOS,0,IERS,6,0,0)
0776 IF<IX .LE. 25)GO TO 34
0777 CALL CHAR<384,0,YPOS,0,IERS,8,0,0)
0778 YPOS = YPOS - 32.0
0779 CBJ
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0779 CBJ+ THIS SECTION HAS BEEN PLACED BELOW AT 38
0780 CBJ
0781 CBJ WRITE OUT RT VALUES FOR SELECTION
0782 CBJ
0783 CB CALL DREAD(NAMEF,20,ILINE)
0784 CB CALL LERS(YPOS,1,VERSN)
0785 CB CALL CHAR(0,0,YPOS,0,ILINE,64,3,3)
0786 CB NIN = 4
0787 CB CALL BLANK(IDATAF,10)
0788 CB CALL INK(2,ITYPE,440,0,YPOS,0, IDATAF,NIN,0,31,0,31,IX,IX)
0789 CB IF(JTYPE .NE. 0)GO TO 31
0790 CB CALL CODE(NIN)
0791 CB READ (IDATAF,*) REFLEC
0792 CB GO TO 32
0793 CB 31 IX = IX/2
0794 CB IF(IX .EQ. 1)REFLEC = 0.0
0795 CB IF(IX .EQ. 3)REFLEC = 0.7
0796 CB IF(IX .EQ. 5)REFLEC = 0.5
0797 CB IF(IX .EQ. 7)REFLEC = 0.3
0798 CB IF(IX .EQ. 9)REFLEC = 0.1
0799 CB IF(IX .EQ. 11)REFLEC = 0.0
0800 CB CALL CODE
0801 CB WRITE (IDATAF,110) REFLEC
0802 CB 32 CALL CHAR(0,0,YPOS,0, IERS,64,0,0)
0803 CB CALL CHAR(40,0,YPOS,0, ILINE,6,0,0)
0804 CB CALL CHAR(80,0,YPOS,0, IDATAF,4,0,0)
0805 CB IF(JTYPE .NE. 0)GO TO 34
0806 CB CALL CODE(4)
0807 CB READ (IDATAF,*) REFLEC
0808 CB 34 YPOS = YPOS - 32.0
0809 CBJ
0810 CBJ
0811 CBJ
0812 CB CALL DREAD(NAMEF,21,ILINE)
0813 CB CALL LERS(YPOS,1,VERSN)
0814 CB CALL CHAR(0,0,YPOS,0, ILINE,46,3,0)
0815 CB CALL CHAR(384,0,YPOS,0, ILINE(25),0,3,0)
0816 CB CALL CHAR(464,0,YPOS,0, ILINE(30),6,0,0)
0817 CB CALL INK(1,ITYPE,0,0,0,0,0,0,0,31,0,31,IX,IX)
0818 CB CALL CHAR(0,0,YPOS,0, ILINE,64,0,0)
0819 CB IF(IX .LE. 25)CALL CHAR(464,0,YPOS,0, IERS,6,0,0)
0820 CB IF(IX .LE. 25)GO TO 36
0821 CB CALL CHAR(384,0,YPOS,0, IERS,8,0,0)
0822 CB YPOS = YPOS - 32.0
0823 CB CALL DREAD(NAMEF,22,ILINE)
0824 CB CALL LERS(YPOS,1,VERSN)
0825 CB CALL CHAR(47,0,YPOS,0, ILINE(4),10,3,0)
0826 CBJ
0827 CBJ
0828 CBJ
0829 CBJ
0830 CBJ
0831 CB CALL BLANK(IDATAF,10)
0832 CB CALL INK(0,ITYPE,144,0,YPOS,0, IDATAF,NIN,0,31,0,31,IX,IX)
0833 CB CALL CODE(NIN)
0834 CB READ (IDATAF,*) BOTLAY
0835 CB CALL CHAR(47,0,YPOS,0, ILINE(4),10,0,0)
0836 CB YPOS = YPOS - 32.0
0837 CB GO TO 38
0838 CB BOTLAY = 0.0
0839 CB
```



ORIGINAL PAGE IS  
OF POOR QUALITY

```
0839      27 CALL CHAR(0.0,YPOS,0,IEFS,64,0,0)
0840      CALL CHAR(0.0,YPOS,0,ILINE,50,0,0)
0841      YPOS = YPOS - 32.0
0842      CBJ*** THIS IS THE END OF THE COMMENTED-OUT SECTION
0843      CBJ
0844      38 CONTINUE
0845      CBJ
0846      CBJ*** INSERTED SECTION ON SURFACE LAYER REFLECTION
0847      CBJ
0848      IF<IRUN,NE.1 GO TO 34
0849      YPOS = YPOS - 32.0
0850      CBJ
0851      CBJ      ENTER SURFACE REFLECTION?
0852      CBJ
0853      CALL DREAD(NAMEF,19,ILINE)
0854      CALL LERS(YPOS,LVERSN)
0855      CALL CHAR(0.0,YPOS,0,ILINE,45,3,0)
0856      CALL CHAR(383.0,YPOS,0,ILINE(25),0,3,0)
0857      CALL CHAR(472.0,YPOS,0,ILINE(30),6,0,0)
0858      CALL INK(1,JTYPE,0,0,0,0,0,0,31,0,31,IX,IY)
0859      CALL CHAR(0.0,YPOS,0,ILINE,64,0,0)
0860      IF<IX,LE,25>REFLEC = 1.0
0861      IF<IX,LE,25>CALL CHAR(464.0,YPOS,0,IEFS,6,0,0)
0862      IF<IX,LE,25>GO TO 34
0863      CALL CHAR(364.0,YPOS,0,IER8,0,0,9)
0864      YPOS = YPOS - 32.0
0865      CBJ
0866      CBJ      WRITE OUT RI VALUES FOR SELECTION
0867      CBJ
0868      CALL DREAD(NAMEF,20,ILINE)
0869      CALL LERS(YPOS,LVERSN)
0870      CALL CHAR(0.0,YPOS,0,ILINE,64,3,0)
0871      NIN = 4
0872      CALL BLANK(IDATAF,10)
0873      CALL INK(2,JTYPE,40.0,YPOS,0,IDATAF,MIN,0,31,0,31,IX,IY)
0874      IF<JTYPE,NE,5>GO TO 31
0875      CALL CODE(MIN)
0876      READ (IDATAF,*) REFLEC
0877      GO TO 32
0878      31 IX = IX/2
0879      IF<IX,EQ,1>REFLEC = 0.8
0880      IF<IX,EQ,3>REFLEC = 0.7
0881      IF<IX,EQ,5>REFLEC = 0.5
0882      IF<IX,EQ,7>REFLEC = 0.3
0883      IF<IX,EQ,9>REFLEC = 0.1
0884      IF<IX,EQ,11>REFLEC = 0.0
0885      CALL CODE
0886      WRITE (IDATAF,10) REFLEC
0887      32 CALL CHAR(0.0,YPOS,0,IEFS,64,0,0)
0888      CALL CHAR(48.0,YPOS,0,ILINE,6,0,0)
0889      CALL CHAR(0.0,YPOS,0,ICNTAF,4,0,0)
0890      IF<JTYPE,NE,6>GO TO 34
0891      34 YPOS = YPOS - 32.0
0892      CBJ
0893      CBJ      DO LOOP ON THE PUN NUMBER
0894      CBJ
0895      CBJ
0896      CBJ
0897      CBJ      40 79 PUNNUM=1,NUMF00
0898      CBJ      42 UP THE FILE NAME FOR THIS PUN. GO UP NUMBER 79
0899      CBJ      43 NAMEF=OUT THE FILE
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0839 C
0900 I = PUNRUM + IFOFF
0901 CALL CODE
0902 WRITE (J,105) I
0903 CALL BZ2(J,FILE(3))
0904 CALL GETIT(ITIME,IDAY,IMONTH,IYEAR)
0905 I = IVHICL + I
0906 WRITE (LUPRNT,200) PUNRUM,(FILE(J),J=1,3),(VIARIES(J,1),J=1,4)
0907 IF(LTIME)WRITE (LUPRNT,201) LTIM,LAUTD(3),LAUTD(4),LDAY,
0908 LMONTH(1),LMONTH(2),LYEAR
0909 WRITE (LUPRNT,202) ITIME,LAUTD(4),IDAY,IMONTH,IYEAR
0910 C
0911 C IF THE DATA IS ON A DISK FILE, READ FROM DISK -- IF IT
0912 C IS ON TAPE, READ IT AS KSC 1965 DATA IN SUBROUTINE KSC65
0913 C
0914 IF(PLACE .NE. 2 .AND. IPLACE .NE. 4) GO TO 39
0915 CALL KSC65(IBUF,IALT,OPTEMP,IFOFF1,IEOF,ITIME,IDAY,IMON,ISHNG,
0916 RUMRUM)
0917 IFOFF1 = I
0918 IF(IEOF .EQ. 1)GO TO 81
0919 IF(IEOF .EQ. 2)GO TO 79
0920 GO TO 48
0921 C
0922 C OPEN THE DATA FILE FOR THIS RUN
0923 C
0924 39 CALL OPEK(IDC,IERR,FILE,0,0,0,272)
0925 IF(IEOF .GE. 0)GO TO 40
0926 WRITE (LUPRNT,203) IERR
0927 GO TO 79
0928 C
0929 C READ THE HEADINGS FROM THE DATA FILE, SETTING UP THE
0930 C APPROPRIATE PARAMETERS
0931 C
0932 40 CALL READ(IDC,IERR,IBUF,40,LEN)
0933 IF(IEOF .GE. 0)GO TO 42
0934 WRITE (LUPRNT,204) IERR
0935 GO TO 79
0936 42 IF(IEOF(1) .NE. T)GO TO 40
0937 WRITE (LUPRNT,205) (IBUF(1),I=1,LEN)
0938 CALL READ(IDC,IERR,IBUF,40,LEN)
0939 IF(IEOF .LT. 0)GO TO 41
0940 IF(IEOF(1).NE.RA .AND. IBUF(1).NE.F)GO TO 43
0941 ISNDF0 = 0
0942 IF(IEOF(1) .EQ. F)ISNDF0 = I
0943 WRITE (LUPRNT,205) (IBUF(1),I=1,LEN)
0944 CALL READ(IDC,IERR,IBUF,40,LEN)
0945 IF(IEOF .LT. 0)GO TO 41
0946 WRITE (LUPRNT,205) (IBUF(1),I=1,LEN)
0947 C
0948 C READ THE SOUNDING/FORECAST TIME
0949 C
0950 CALL READ(IDC,IERR,IBUF,9,LEN)
0951 IF(IEOF .LT. 0)GO TO 41
0952 CALL CODE(18)
0953 READ (IBUF,104) IESTIME,ISDAY,ISHNG(1),ISHNG(2),IYEAR
0954 C
0955 C CHANGE TO EST OR EDT DEPENDING ON LAUNCH TIME
0956 C
0957 IESTIME = IESTIME - 500
0958 IESTIME = IESTIME + IESTIME - 500
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0959 IF(LAUNIT<4) .NE. 2)ST)ISTIME = ISTIME + 1000
0960 IF(ISTIME .GT. 0)GO TO 44
0961 ISTIME = 2400 + ISTIME
0962 ISDAY = ISDAY - 1
0963 C
0964 C WRITE OUT THE NEXT LINE OF THE HEADER
0965 C
0966 44 CALL READF(IDC8, IERR, IBUF, 40, LEN)
0967 IF(IERR .LT. 0)GO TO 41
0968 WRITE (LUPRINT, 205) (IBUF(I), J=1, LEN)
0969 C
0970 C WRITE OUT THE SOUNDING/FORECAST TIME
0971 C
0972 WRITE (LUPRINT, 206) ISTIME, ITIMEZ, LAUNIT<4>, ISDAY, ISMON<1>,
0973 ISMON<2>, ISYEAR
0974 C
0975 C FIND THE FIRST DATA POINT WITH AN ALTITUDE OF 10 FEET
0976 C OR ABOVE
0977 C
0978 45 CALL READF(IDC8, IERR, IBUF, 40, LEN)
0979 IF(IERR .LT. 0)GO TO 41
0980 CALL BZ<IBUF, J)
0981 IF(J.LT.ZER00 .OR. J.GT.ZEK01)GO TO 45
0982 CALL CODE<2>LEN)
0983 READ (IBUF, 105) IALT<1>, IDIR<1>, SPEED<1>, TEMP<1>, DPTEMP<1>,
0984 PRESS<1>, SUPDEN
0985 IF(IALT<1> .LT. 10)GO TO 45
0986 C
0987 C TRY TO FIND A TOTAL OF 30 DATA POINTS WITH ALTITUDES
0988 C BETWEEN 20 FT AND 10,000 FT INCLUSIVE
0989 C
0990 NUM = 1
0991 DO 47 I=2,30
0992 46 CALL READF(IDC8, IERR, IBUF, 40, LEN)
0993 IF(IERR.LT.0 .AND. IERR.NE.-12)GO TO 41
0994 IF(LEN .EQ. -1)GO TO 48
0995 CALL BZ<IBUF, J)
0996 IF(J.LT.ZER00 .OR. J.GT.ZEK01)GO TO 46
0997 CALL CODE<2>LEN)
0998 READ (IBUF, 105) IALT<1>, IDIR<1>, SPEED<1>, TEMP<1>, DPTEMP<1>,
0999 PRESS<1>
1000 IF(IALT<1> .LT. 20 .OR. IALT<1> .GT. 10000)GO TO 46
1001 47 NUM = I
1002 C
1003 C ZERO OUT THE REMAINING ELEMENTS OF THE ARRAYS
1004 C
1005 NUM1 = NUM + 1
1006 IF(NUM1 .GT. 30)GO TO 51
1007 DO 49 I=NUM1,30
1008 ALI<I> = 0.0
1009 IDIR<I> = 0
1010 SPEED<I> = 0.0
1011 TEMP<I> = 0.0
1012 DPTEMP<I> = 0.0
1013 49 PRESS<I> = 0.0
1014 C
1015 C CONVERT TO METRIC UNITS
1016 C
1017 51 DO 52 I=1,NUM
DATA
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
1019 C      54 SPEED(I) = 0.515 * SPEED(I)
1020 C
1021 C      SORT ALL THE DATA POINTS SO THEY APPEAR IN ASCENDING
1022 C      ORDER OF ALTITUDE
1023 C
1024 C      NUM1 = NUM - 1
1025 C      DO 58 I=1,NUM1
1026 C      JJ = NUM - I
1027 C      DO 57 J=1, JJ
1028 C      JI = J + 1
1029 C      IF(ALT(J) .LE. ALT(JI))GO TO 57
1030 C      ARG = ALT(J)
1031 C      ALT(J) = ALT(JI)
1032 C      ALT(JI) = ARG
1033 C      IARG = IDIR(J)
1034 C      IDIR(J) = IDIR(JI)
1035 C      IDIR(JI) = IARG
1036 C      ARG = SPEED(J)
1037 C      SPEED(J) = SPEED(JI)
1038 C      SPEED(JI) = ARG
1039 C      ARG = TEMP(J)
1040 C      TEMP(J) = TEMP(JI)
1041 C      TEMP(JI) = ARG
1042 C      ARG = DTEMP(J)
1043 C      DTEMP(J) = DTEMP(JI)
1044 C      DTEMP(JI) = ARG
1045 C      ARG = PRESS(J)
1046 C      PRESS(J) = PRESS(JI)
1047 C      PRESS(JI) = ARG
1048 C      *7 CONTINUE
1049 C      58 CONTINUE
1050 C
1051 C      CALCULATE THE POTENTIAL TEMPERATURE
1052 C
1053 C      DO 62 I=1,NUM
1054 C      62 PTEMP(I) = (TEMP(I) + 273.16) * ((1000.0/PRESS(I))**.288)
1055 C
1056 C      WRITE THE HEADER FOR SOUNDING OR FORECAST
1057 C
1058 C      IF(ISHMFO .EQ. 1)GO TO 64
1059 C      WRITE (LUPRINT,207)
1060 C      GO TO 65
1061 C      64 WRITE (LUPRINT,208)
1062 C
1063 C      WRITE THE SURFACE DENSITY AND ALL THE DATA POINTS
1064 C
1065 C      65 WRITE (LUPRINT,209) SURDEN
1066 C      WRITE (LUPRINT,210)
1067 C      DO 68 I=1,NUM
1068 C      IALTF = 3.261 * ALT(I) + 0.5
1069 C      IALTM = ALT(I) + 0.5
1070 C      APTEMP = PTEMP(I) - 273.16
1071 C      68 WRITE (LUPRINT,211) I,IALTF,IALTM,IDIR(I),SPEED(I),TEMP(I),
1072 C      APTEMP,DTEMP(I),PRESS(I)
1073 C
1074 C      CLOSE THE DATA FILE
1075 C
1076 C      CALL CLOSE(LUCF)
1077 C
```



ORIGINAL PAGE IS  
OF POOR QUALITY

```
1139 CC      CALL CLEAR
1140      IF(LVERSH) CALL CLEAR
1141      CALL INGRAF
1142      C
1143      C      OF ALL SEGMENTS -- IF ERRORS ENCOUNTERED, WRITE OUT A MESSAGE
1144      C      AND TERMINATE REEDS
1145      C
1146      DO 94 I=1,NUMSEC
1147      CALL IGRPD(NAMP(1,1),IERR)
1148      IF(IERR .GE. 0)GO TO 94
1149      WRITE (LU,213) IERR
1150      STOP
1151      94 CONTINUE
1152      C
1153      C      TERMINATE REEDS
1154      C
1155      C      STOP
1156      C
1157      C      END
1158      C      END*
```

1-2

ORIGINAL PAGE IS  
OF POOR QUALITY

```

SPEED T-0003 IS ON CR0003 USING 00141 BLKS K-6000
      FTH4X,L
      SUBROUTINE PWDIS(NAME, JJ),
      -----
      THIS SUBROUTINE READS AND WRITES COMMON AREAS TO DISC
      TO BE UTILIZED BY THE VARIOUS PWD PROGRAMS.
      -----
      >>> COMMON AREAS
      >>> MATH PARAMETERS
      COMMON PI,PI0VR2,P143,TU0PI,SOR2PI,DEGRAD,RADDEG,IV2
      >>> VEHICLE PARAMETERS
      COMMON VPAR(10),SIGHCL
      >>> OPTION PARAMETERS
      COMMON YPOS,YES,ISMOFO,IVNICL,IRUM,ICALC,IPLACE,ITIMEZ,ICOTMP,
      HAMPFH,LU,NUM,HIGHLY,NFDRH,LUPRHT,ITDU,IPL1,IPL2,IPL3
      INTEGER YES
      >>> MET DATA
      COMMON ALT(30),IDIR(30),SPEED(30),TEMP(30),PRESS(30),PTEMP(30),
      SURDEN,FILE(3)
      INTEGER FILE
      >>> CALCULATION TIME
      COMMON ITIME,IDAY,IMONTH(2),IYEAR
      >>> SOUNDING TIME
      COMMON ISTIME,ISDAY,ISMOIK(2),ISYEAR
      >>> LAUNCH TIME
      COMMON LTIME,LTIM,LDAY,LMONTH(2),LYEAR,LAUNTD(10)
      LOGICAL LTIME
      >>> CLOUD STABILIZATION AND LAYER PARAMETERS
      COMMON STBLT,STMT,STBZD,STBPG,STBTH,CLDRAD,PISTIM(30),BOTLAY,
      TOPLAY,CALHT,LAYTOP,LAYBOT,REFLEC,IRIZ
      >>> DIFFUSION COEFFICIENTS
      COMMON SIGX,SIGY,SIGZ,SIGAP,XDIST,YDIST,EXPZ,SOSIGZ,SIGY,SIGAL,
      CLUNG,MVSPD,RVDIR,SIGAZ
      >>> CONCENTRATION AND DOSAGE VALUES AND RANGES
      COMMON CONMX,CONCFX,DOSTMX,DOSPK,ENGSMC,ENGSRD
      >>> EQUIVALENCE STATEMENT FOR VEHICLE PARAMETERS
      EQUIVALENCE (OC1,VPAR(1)),(OC2,VPAR(2)),(UL3,VPAR(3)),
      (OT1,VPAR(4)),(OT2,VPAR(5)),(OT3,VPAR(6)),
      (AA,VPAR(7)),(BB,VPAR(8)),(CC,VPAR(9)),
      (HEATH,VPAR(10)),(HEATH,VPAR(11)),(HEATH,VPAR(12)),
      (PHCL,VPAR(13)),(PCO,VPAR(14)),(PCO,VPAR(15)),
      (PAL203,VPAR(16)),(PIN0,VPAR(17)),(GAMMA2,VPAR(18))
      INTEGER (OC1(14)),(OC2(5))
  
```

9 15  
17 27  
19 20  
14 335  
5 5  
5 5  
14 14  
43 9  
1  
1

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0053 DIMENSION NAME(3), ISIZE(2)
0060 EQUIVALENCE (OBUF(1),PI)
0061
0062 CBJ *** ISIZE(1) = 0 BLOCKS EXPECTED TO BE WRITTEN READ DURING RUN
0063 CBJ *** ISIZE(2) = 0 WORDS IN THE COMMON BLOCK TO BE TRANSFERRED
0064 DATA ISIZE(3),557/
0065
0066 CBJ *** IF THE SCRATCH DATA FILE (.I.,-REED2) ISN'T THERE, CREATE IT
0067 CBJ *** G E U A R E I I BE SURE FILE IS PURGED BEFORE STARTING RUN I I
0068 CALL OPEN(ODCB, IERR, NAME, 0)
0069 IFC (IERR, LT. 0) .AND. (IERR, NE. -6) ) WRITE(6, 770) IERR
0070 FORMAT(' *** ATTEMPT TO OPEN, IERR =', I4)
0071 IFC (IERR, LT. 0)
0072 CALL CREAT(ODCB, IERR, NAME, ISIZE, 3, 0, 0, 144)
0073 IFC (IERR, LE. 0) WRITE(6, 799) IERR
0074 FORMAT(' *** CREATE/OPEN, IERR =', I4)
0075
0076 CBJ *** READ OR WRITE, AS PER ORDER
0077 IFC (J, EQ. 1) CALL WRITF(ODCB, IERR, OBUF, ISIZE(2))
0078 IFC (J, EQ. 0) CALL READF(ODCB, IERR, OBUF, ISIZE(2))
0079 CALL CLOSE(ODCB, IERR)
0080 RETURN
0081 END
0082 SUBROUTINE KSC65( IBUF, IALT, DPTIME, IUAHT, IEOF, ITING, IDAYG, IMING,
0083 ISHDC, RUNNUM )
0084 C
0085 C -----
0086 C
0087 C THIS SUBROUTINE READS IN DATA FOR THE REED DIFFUSION
0088 C MODEL FROM MAG TAPE IN KSC 1965 FORMAT
0089 C -----
0090 C
0091 C
0092 C >>> COMMON AREAS
0093 C
0094 C
0095 C >>> MATH PARAMETERS
0096 C COMMON PI, FIDVR2, FI43, TUOPI, SOR2PI, DEGRAD, PADDEC, IV2
0097 C
0098 C >>> VEHICLE PARAMETERS
0099 C COMMON VPAR(18), SIGHCL
0100 C
0101 C >>> OPTION PARAMETERS
0102 C COMMON YPOS, YES, IZIMFO, IWHICL, IPUN, ICALC, IPLACE, ITIMEZ, ICOMPF,
0103 MURKON, LU, NUM, MOHLY, MFORM, LDFRHT, ITDU, IPL1, IPL2, IPL3
0104 C
0105 C >>> MET DATA
0106 C COMMON ALIC(30), IDIR(30), SPEED(30), TEMP(30), PRESS(30), PTEMP(30),
0107 C SURDEN, FILE(3)
0108 C
0109 C >>> CALCULATION TIME
0110 C COMMON ITIME, IDAY, IMONTH(2), IYEAR
0111 C
0112 C >>> SOUNDING TIME
0113 C COMMON ISTIME, ISDAY, ISMON(2), ISYEAR
0114 C
0115 C >>> LAUNCH TIME
0116 C
```



ORIGINAL PAGE IS  
OF POOR QUALITY

```
0119 COMMON LTIME, LTIME, LMONTH(2), LYEAR, LAINTD(16),
0120 LOGICM, LTIME
0121 C
0122 C >>> CLOUD STABILIZATION AND LAYER PARAMETERS
0123 COMMON STBLT, STBLT, STBZTD, STBPHG, STBTIM, CLDOPD, RSTIMC(30), ROTLAY,
0124 TOPLAY, CALHT, LAYTOP, LAYBOT, REFLEC, IPTZ
0125 C
0126 C >>> DIFFUSION COEFFICIENTS
0127 COMMON SIGX0, SIGX, SIGZ0, SIGAP, XDIST, YDIST, EXPZ, SOSICZ, SIGY, SIGAL,
0128 CLDLNG, AVSFD, AVDIR, SIGAZ
0129 C
0130 C >>> CONCENTRATION AND DOSAGE VALUES AND RANGES
0131 COMMON CONMAX, CONCPK, DOSMAX, DOSPK, RANGSMC, RANGSMO
0132 C
0133 C >>> EQUIVALENCE STATEMENT FOR VEHICLE PARAMETERS
0134 EQUIVALENCE (QCI, VPAR(1)), (QC2, VPAR(2)), (QC3, VPAR(3)),
0135 (QTI, VPAR(4)), (QT2, VPAR(5)), (QT3, VPAR(6)),
0136 (QA, VPAR(7)), (BB, VPAR(8)), (CC, VPAR(9)),
0137 (HEATN, VPAR(10)), (HEATA, VPAR(11)), (HEATA, VPAR(12)),
0138 (PACL, VPAR(13)), (PCO, VPAR(14)), (PCO2, VPAR(15)),
0139 (PAL203, VPAR(16)), (PNO, VPAR(17)), (GAMMAX, VPAR(18))
0140 C
0141 C** DIMENSION STATEMENTS
0142 C
0143 C DIMENSION IMONG(5,2), ITIMC(5), IDAYC(5)
0144 C
0145 C** TYPE STATEMENTS
0146 C
0147 C INTEGER RUNNUM
0148 C
0149 C INPUT FORMAT STATEMENTS
0150 C
0151 1000 FORMAT (40A2)
0152 1001 FORMAT (14,3X12,1X2,A1,1X14)
0153 1002 FORMAT (1X16,3X13,5XF3.0,2XF5.1,3XF5.1,3XF6.1,15XF6.1)
0154 C
0155 C OUTPUT FORMAT STATEMENT
0156 C
0157 2000 FORMAT ('05X"TIME: "14,1X2,A1,2,4X"DATE: "12,1X2,A1,1X14)
0158 C
0159 C DIMENSION STATEMENT
0160 C
0161 C DIMENSION IBUF(1), IALT(1), DTEMP(1)
0162 C
0163 C INITIALIZE THE COUNTER FOR THE NUMBER OF SETS OF DATA TO 0
0164 C
0165 C ICONT = 0
0166 C
0167 C READ DATA FROM TAPE
0168 C
0169 C 4 READ (0,1000) (IBUF(I),I=1,40)
0170 C
0171 C IF AN EOF ON TAPE, SET THE EOF FLAG, AND RETURN
0172 C
0173 C CALL EXEC(13,8,IEGTS)
0174 C IEOF = IAND(ISHIF(IEGTS,-7),1)
0175 C IFCIEOF .EQ. 1) RETURN
0176 C
0177 C KEEP READING UNTIL THE STAMPO LEVEL DATA IS FOUND
0178 C
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0179 C IF<IBUF<2>.NE.2HST>GO TO 4
0180 C 7 READ<0,1000><IBUF<1>,1=1,40>
0181 C IF<IBUF<1>.NE.2MCA.OR.IBUF<2>.EQ.2HST>GO TO 7
0182 C
0183 C READ THE SOUNDING/FORECAST TIME
0184 C
0195 C READ<0,1001>ISTIME,ISDAY,ISMOK<1>,ISMOK<2>,ISYEAR
0196 C
0197 C** CHECK FOR SPECIFIED SOUNDING TIME AND DATE
0198 C
0199 C IF<ISMOG.EQ.0>GO TO 0
0200 C IF<ISTIME.EQ.1TING<RUNNUM>>.AND.ISDAY.EQ.IDAYC<RUNNUM>>.AND.
0201 C ISMOK<1>.EQ.IMONG<RUNNUM,1>.AND.ISMOK<2>.EQ.IMDNG<RUNNUM,2>>
0202 C GO TO 0
0203 C GO TO 4
0204 C
0205 C CHANGE TO EST OR EDT DEPENDING ON LAUNCH TIME
0206 C
0207 C CONTINUE
0208 C ISTEME = ISTEME - 300
0209 C IF<IPLACE.EQ.1>ISTIME = ISTEME - 300
0210 C IF<LAUNTD<4>.NE.2HST>ISTIME = ISTEME + 100
0211 C IF<ISTIME.GT.0>GO TO 11
0212 C ISTEME = 2400 + ISTEME
0213 C ISDAY = ISDAY - 1
0214 C
0215 C FIND THE KEY WORD ALTITUDE
0216 C
0217 C 11 READ<0,1065><IBUF<1>,1=1,40>
0218 C IF<IBUF<2>.EQ.2HST>GO TO 7
0219 C IF<IBUF<1>.NE.2MCA>GO TO 11
0220 C
0221 C LIMIT DATA TO 30 POINTS -- READ THE STANDARD LEVEL DATA
0222 C DO 19 I=1,30
0223 C 15 READ<0,1002>IALT<I>,IDIR<I>,SPEED<I>,TEMP<I>,OPTEMP<I>,PRESS<I>,
0224 C SURGN
0225 C IF<SPEED<I>.EQ.999.0.OR.IDIR<I>.EQ.999>GO TO 15
0226 C IF<IDIR<I>.EQ.36.3>IDIR<I> = 0
0227 C IF<I.EQ.1>SURDEN = SURDN
0228 C IF<IALT<I>.GT.10000>GO TO 22
0229 C 19 CONTINUE
0230 C 22 NUM = I
0231 C IF<NUM.GT.30>GO TO 34
0232 C
0233 C FIND THE KEY WORD MANDATORY
0234 C
0235 C 25 READ<0,1000><IBUF<1>,1=1,40>
0236 C IF<IBUF<2>.EQ.2HST>GO TO 7
0237 C IF<IBUF<10>.NE.2MOR.AND.IBUF<15>.NE.2HOR>GO TO 25
0238 C READ<0,1000>I
0239 C
0240 C LIMIT DATA TO 30 POINTS -- READ THE MANDATORY LEVEL DATA
0241 C
0242 C DO 29 I=NUM,30
0243 C 27 READ<0,1002>IALT<I>,IDIR<I>,SPEED<I>,TEMP<I>,OPTEMP<I>,PRESS<I>
0244 C IF<SPEED<I>.EQ.999.0.OR.IDIR<I>.EQ.999>GO TO 27
0245 C IF<IDIR<I>.EQ.36.0>IDIR<I> = 0
0246 C IF<IALT<I>.GT.10000>GO TO 34
0247 C 29 CONTINUE
0248 C
```

```

0239 C      NUM IS THE NUMBER OF DATA POINTS
0240 C
0241 C      34 NUM = I - 1
0242 C
0243 C      INCREMENT THE COUNTER -- IF THIS IS THE SET OF DATA DESIRED,
0244 C      WRITE OUT THE SOUNDING/FORECAST TIME -- OTHERWISE GET THE NEXT
0245 C      SET
0246 C
0247 C      IGOY = IGOY + 1
0248 C      IF(I GOY .LT. I WANT) GO TO 4
0249 C
0250 C      WRITE OUT THE SOUNDING/FORECAST TIME
0251 C
0252 C      WRITE (LUPPNT,2006) ISTIME,ITIMEZ,LAUNTD(4),15DH/,15MONK1),
0253 C      ISMONK2),15YEAR
0254 C
0255 C      THERE MUST BE 5 OR MORE DATA POINTS FOR THIS TO BE A VALID SET
0256 C      OF DATA -- IF THERE IS NOT, RETURN WITH IEOF=2
0257 C
0258 C      IFCNUM .GE. 5) RETURN
0259 C      IEOF = 2
0260 C      RETURN
0261 C
0262 C      END OF KSC65
0263 C
0264 C
0265 C      END
0266 C
0267 C      SUBROUTINE GETD(TIME, IDAY, IMONTH, IYEAR)
0268 C
0269 C
0270 C
0271 C
0272 C
0273 C
0274 C
0275 C
0276 C
0277 C
0278 C
0279 C
0280 C
0281 C
0282 C
0283 C
0284 C
0285 C
0286 C
0287 C
0288 C
0289 C
0290 C
0291 C
0292 C
0293 C
0294 C
0295 C
0296 C
0297 C
0298 C

```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0299 IF(4+1.EQ.1YEAR)DAYMO(2) = 29
0300 C
0301 C CONVERT THE JULIAN DAY INTO A MONTH AND A DAY
0302 C
0303 IDAY = IT(5)
0304 DO 7 I=1,12
0305 IDAY = IDAY - DAYMO(I)
0306 IF(IDAY.LE.0)GO TO 12
0307 7 CONTINUE
0308 12 IDAY = IDAY + DAYMO(I)
0309 MONTH(1) = MONTHS(I,1)
0310 MONTH(2) = MONTHS(2,I)
0311 C
0312 C RETURN TO THE CALLING PROGRAM
0313 C
0314 C RETURN
0315 C
0316 C END OF GETD
0317 C
0318 C
0319 C SUBROUTINE B2Z(IA,IB)
0320 C
0321 C -----
0322 C - THIS SUBROUTINE CHANGES BLANK FILLED WORDS TO ZEROS. -
0323 C -
0324 C -
0325 C -----
0326 C
0327 IB = IAND(IA,1774000)
0328 IF(1B.EQ.0200000)IB = 0300000
0329 IC = IAND(IA,000377B)
0330 IF(1C.EQ.000400)IC = 0000600
0331 IB = IOR(1B,IC)
0332 RETURN
0333 END
0334 SUBROUTINE DREAD(NAMEF,LINR,ILINE)
0335 C
0336 C -----
0337 C - THIS SUBROUTINE READS A SPECIFIED QUESTION FROM A SPECIFIED
0338 C - QUESTION FILE, THE QUESTION IS THEN DISPLAYED UPON THE
0339 C - FLASCOPE FOR PROCESSING.
0340 C -
0341 C -
0342 C -----
0343 C
0344 DIMENSION NAMEF(3),IDCB(14),IBUF(40),ILINE(32)
0345 CALL OPEN(IDCB,IERR,NAMEF,0)
0346 LOOP = LINR - 1
0347 DO 10 I=1,LOOP
0348 CALL BLANK(1BUF,40)
0349 CALL READF(IDCB,IERR,IBUF)
0350 CONTINUE
0351 CALL BLANK(1BUF,40)
0352 CALL READF(IDCB,IERR,IBUF)
0353 CALL CODE
0354 READ(1BUF,100) (ILINE(I),I=1,32)
0355 FORMAT(32A2)
0356 CALL CLOSE(IDCB,IERR)
0357 RETURN
0358 END
0359 C
0360 C
0361 C
0362 C
0363 C
0364 C
0365 C
0366 C
0367 C
0368 C
0369 C
0370 C
0371 C
0372 C
0373 C
0374 C
0375 C
0376 C
0377 C
0378 C
0379 C
0380 C
0381 C
0382 C
0383 C
0384 C
0385 C
0386 C
0387 C
0388 C
0389 C
0390 C
0391 C
0392 C
0393 C
0394 C
0395 C
0396 C
0397 C
0398 C
0399 C
0400 C
0401 C
0402 C
0403 C
0404 C
0405 C
0406 C
0407 C
0408 C
0409 C
0410 C
0411 C
0412 C
0413 C
0414 C
0415 C
0416 C
0417 C
0418 C
0419 C
0420 C
0421 C
0422 C
0423 C
0424 C
0425 C
0426 C
0427 C
0428 C
0429 C
0430 C
0431 C
0432 C
0433 C
0434 C
0435 C
0436 C
0437 C
0438 C
0439 C
0440 C
0441 C
0442 C
0443 C
0444 C
0445 C
0446 C
0447 C
0448 C
0449 C
0450 C
0451 C
0452 C
0453 C
0454 C
0455 C
0456 C
0457 C
0458 C
0459 C
0460 C
0461 C
0462 C
0463 C
0464 C
0465 C
0466 C
0467 C
0468 C
0469 C
0470 C
0471 C
0472 C
0473 C
0474 C
0475 C
0476 C
0477 C
0478 C
0479 C
0480 C
0481 C
0482 C
0483 C
0484 C
0485 C
0486 C
0487 C
0488 C
0489 C
0490 C
0491 C
0492 C
0493 C
0494 C
0495 C
0496 C
0497 C
0498 C
0499 C
0500 C
```



ORIGINAL PAGE IS  
OF POOR QUALITY

```

0419 2H5 ,2H D,2HR,2H ,2H54,2H F,2HT,2H ,2H ,2H
DATA LINES/2H ,2H ,2H ,2H ,2H ,2H ,2H ,2H ,2H ,2H ,2H
0420 2HMB,2HR ,2H ,2H ,2H ,2H ,2H ,2H ,2H ,2H ,2H
0421 2HSP,2HD ,2H ,2H ,2H ,2H ,2H ,2H ,2H ,2H ,2H
0422 2HSP,2HD ,2H ,2H ,2H ,2H ,2H ,2H ,2H ,2H ,2H
0423 2HSP,2HD ,2H ,2H ,2H ,2H ,2H ,2H ,2H ,2H ,2H
0424 2H D,2HP ,2H ,2H ,2H ,2H ,2H ,2H ,2H ,2H ,2H
0425 2HM ,2H D,2HEV,2H ,2HT,2H ,2HRH,2H ,2H ,2H
0426 2H ,2H ,2H ,2H ,2H ,2H ,2H ,2H ,2H ,2H ,2H
0427 2H ,2H ,2H ,2H ,2H ,2H ,2H ,2H ,2H ,2H ,2H
0428 2H95,2H ,2H ,2H ,2H ,2H ,2H ,2H ,2H ,2H ,2H
0429 2H ,2H(4,2H92,2H) /
0430 DATA LEVEL,12,54,162,204,295,354,492/
0431 DATA LEVEL,2/6,54/
0432 DATA NAME,2/HTD,2/IME,2/SPS/
0433 C
0434 C
0435 C
0436 100 FORMAT(/" THE NAME OF THE FILE TO BE USED (I11AAAAAA) _")
0437 101 FORMAT(" IERR =",I4)
0438 102 FORMAT(/" THE TEST NUMBER (I11XXXX) _")
0439 103 FORMAT(I1A2)
0440 104 FORMAT(4A2,1X,16,1X)
0441 105 FORMAT(I2A2)
0442 106 FORMAT(58A2)
0443 108 FORMAT(34A2)
0444 109 FORMAT(/" IS THE DATA ENTERED CORRECTLY SO FAR? (L1JJY=YES(L1D0"
      " ,N=NO,S=STOP TOWER) _")
0445 110 FORMAT(" THE YEAR (I11XX) _")
0446 111 FORMAT(" THE INTEGER REPRESENTING THE MONTH (I11XX) _")
0447 112 FORMAT(" THE DAY OF THE MONTH (I11XX) _")
0448 113 FORMAT(" THE TIME (2) (I11XXXX) _")
0449 114 FORMAT(" THE TIME INTERVAL (MIN I11XX) _")
0450 115 FORMAT(/" THE TOWER NUMBER (I11XX) _")
0451 116 FORMAT(5X,"WIND DIRECTION (DEC I11XX) _")
0452 117 FORMAT(5X,"WIND SPEED (KTS I11XX) _")
0453 118 FORMAT(5X,"GUST SPEED (KTS I11XX) _")
0454 119 FORMAT(/" THE",1X,12,1X,"FOOT TEMPERATURE (DEC F I11XX) _")
0455 120 FORMAT(" THE",1X,12,1X,"FOOT DEW POINT (DEC F I11XX) _")
0456 121 FORMAT(/" THE LAPSE RATE (DEC F I11XX.X) _")
0457 122 FORMAT(" THE 5 PPM DISTANCE (MI I11XX.X) _")
0458 123 FORMAT(" THE 25 PPM DISTANCE (MI I11XX.X) _")
0459 124 FORMAT(" THE STANDARD DEVIATION OF THE WIND AZIMUTH ANGLE"
      " (DEC I11XX.X) _")
0460 125 FORMAT(" THE RELATIVE HUMIDITY AT 54 FEET (2 I11XX) _")
0461 126 FORMAT(2(I2,2X),1X,13,2X,12,2X,12,2X,12,2X,12,1X,13),
      " 2(I2X,12),1X,13,2(I1X,13),3X,5,4,1,2X,2(F4,1,1X),F4,1,2(I1X,13))
0462 127 FORMAT(/" DATA FILE",1X,3A2,1X," IS READY")
0463 128 FORMAT(36X,3(I3,2X,12,2X,12,1X),42X)
0464 129 FORMAT(/" IS THERE DATA FOR ANOTHER VEHICLE USING THE SAME TIME?"
      " ,Y=YES,(L1JJY=IN(L1D0) _")
0465 130 FORMAT(/1X,13,1X,"FOOT LEVEL DATA")
0466 131 FORMAT(/" THE DATA ENTERED IS: YEAR MONTH DAY TIME INT/"
      " 24X,12,4X,12,2X,12,1X,14,1X,13)
0467 132 FORMAT( 5X,13,8X,13,5X,12,5X,12)
0468 133 FORMAT(/" THE TEMPERATURES AND DEW POINTS ENTERED ARE:"
      " 5X"ALTITUDE TEMPERATURE AND DEW POINTS ENTERED ARE:"
      " 16X,13,5X,13,7X,13/16X,13,5X,13,7X,13)
0469 134 FORMAT(/" THE DATA ENTERED IS:"
      " 5X," LAPSE RATE 5PPH 25PPH SIGMA PH/"
      " 12X,F4,1,2X,F4,1,3X,F4,1,3X,F4,1,12,13)
0470
0471
0472
0473
0474
0475
0476
0477
0478

```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0479 135 FORMAT(/" FOR TOWER NUMBER",IX,I3,I3,IX,"THE DATA ENTERED IS:"/
0480 5X,"ALT DIRECTION SPEED GUSTS"/)
0481 136 FORMAT(" ENTER MODE: {&dJO=OPERATIONAL,&dO,P=PRODUCTION"
0482 " ,R=RESEARCH "})
0483 137 FORMAT(/" ENTER THE VEHICLE {&dJS=SHUTTLE,&dO,T=TITAN,D=DELTA " /)
0484 138 FORMAT(/" DO YOU WANT A DETAILED DATA FILE? (Y=YES, {&dDN=NO,&dO})"
0485 " " /)
0486 139 FORMAT(5X,I3,IX,I3,5X,I2)
0487 140 FORMAT(/" FOR TOWER NUMBER",IX,I3,I3,IX,"THE DATA ENTERED IS:"/
0488 5X,"ALT DIRECTION SPEED "/)
0489 141 FORMAT(/" THE TEMPERATURES ENTERED ARE,"/5X,"ALTITUDE"
0490 " TEMPERATURE"/10X,I3,9X,I3/10X,I3,9X,I3)
0491 142 FORMAT(/" IS THERE DATA FOR ANOTHER TOWER NUMBER USING THE SAME"
0492 " TIME? (Y=YES, {&dJH=NO,&dO}) " /)
0493 143 FORMAT(/" ENTER DATA FOR WIND SYSTEM TOWER NUMBER",IX,I3)
0494 200 FORMAT(3A2)
0495 201 FORMAT(A1)
0496 202 FORMAT(I5)
0497 209 FORMAT(A2)
0498 210 FORMAT(I2)
0499 211 FORMAT(I4)
0500 213 FORMAT(I3)
0501 221 FORMAT(F4,1)
0502 C
0503 C SET THE VARIABLE 'MODE' IDENTIFYING THE RUNNING MODE, OPERATION, RE-
0504 C SEARCH, OR PRODUCTION. IF IN OPERATION MODE, ENTER ONLY THE DATA
0505 C THAT IS ABSOLUTELY NECESSARY...IF IN THE PRODUCTION MODE, ALLOW
0506 C THE USER TO DECIDE WHETHER OR NOT TO ENTER ALL OR ONLY ESSENTIAL
0507 C DATA...IF IN RESEARCH MODE, ENTER ALL DATA. THE VARIABLE, 'IQIK'
0508 C TESTS FOR ALL OR ONLY ESSENTIAL DATA INPUT.
0509 C
0510 IQIK = 1
0511 IF(IIRUM.EQ.2) GO TO 1
0512 IQIK = 0
0513 IF(IIRUM.EQ.1) GO TO 24
0514 WRITE (LU,138)
0515 READ (LU,201) IANS
0516 IF(IANS.EQ.1) IQIK = 1
0517 IF(IQIK.EQ.1) GO TO 1
0518 C
0519 C GET THE NAME OF THE DATA FILE TO BE USED OR CREATED
0520 C --LET THE FILE NAMED 'TOWERS' BE CREATED IF ONLY ESSENTIAL DATA
0521 C IS BEING ENTERED.
0522 C
0523 24 WRITE (LU,100)
0524 READ (LU,200)IAMS2
0525 C
0526 C OPEN OR CREATE THE DATA FILE NEEDED
0527 C
0528 1 CALL OPEN(MCDB,IERR,IAMS2,0)
0529 IF(IERR.LT.0) CALL CREAT(MCDB,IERR,IAMS2,4,4,0,0,144)
0530 IF(IERR.LT.0) WRITE (LU,101)IERR
0531 C
0532 C WRITE THE TEST NUMBER TO THE FILE -- LET THE TEST NUMBER DEFAULT
0533 C TO 00999 IF ENTERING ESSENTIAL DATA ONLY.
0534 C
0535 IF(IQIK.EQ.1) NTEST = 00999
0536 IF(IQIK.EQ.1) GO TO 2
0537 WRITE (LU,102)
0538 READ (LU,202)NTEST
0539 C
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0539 2 CALL CODE
0540 WRITE(NBUF,104)LINE1,NTEST
0541 CALL CODE
0542 CALL WRITF(NDCB,IERR,NBUF,7)
0543 IF(IERR.LT.0) WRITE(LU,101)IERR
0544 C
0545 C WRITE WIND SYSTEM TOWER DATA HEADING
0546 C
0547 CALL CODE
0548 WRITE(NBUF,103)LINE2
0549 CALL CODE
0550 CALL WRITF(NDCB,IERR,NBUF,11)
0551 IF(IERR.LT.0) WRITE(LU,101) IERR
0552 C
0553 C WRITE THE CAPE CANAVERAL HEADING
0554 C
0555 CALL CODE
0556 WRITE(NBUF,105)LINE3
0557 CALL CODE
0558 CALL WRITF(NDCB,IERR,NBUF,12)
0559 IF(IERR.LT.0) WRITE(LU,101) IERR
0560 C
0561 C WRITE THE FIRST LINE OF THE HEADER FOR THE DATA COLUMNS
0562 C
0563 CALL CODE
0564 WRITE(NBUF,106)LINE4
0565 CALL CODE
0566 CALL WRITF(NDCB,IERR,NBUF,50)
0567 IF(IERR.LT.0) WRITE(LU,101) IERR
0568 C
0569 C WRITE THE SECOND LINE OF THE DATA COLUMNS HEADINGS
0570 C
0571 CALL CODE
0572 WRITE(NBUF,106)LINE5
0573 CALL CODE
0574 CALL WRITF(NDCB,IERR,NBUF,50)
0575 IF(IERR.LT.0) WRITE(LU,101) IERR
0576 C
0577 C WRITE THE THIRD LINE OF THE HEADINGS FOR THE DATA COLUMNS
0578 C
0579 CALL CODE
0580 WRITE(NBUF,108)LINE6
0581 CALL CODE
0582 CALL WRITF(NDCB,IERR,NBUF,34)
0583 IF(IERR.LT.0) WRITE(LU,101) IERR
0584 C
0585 C ENTER THE DATE, TIME, AND TIME INTERVAL, LET THE TIME AND DATE
0586 C DEFAULT TO 99 AND 9999 IF ONLY ESSENTIAL DATA IS BEING ENTERED.
0587 C
0588 IF(IQIK.EQ.1) GO TO 4
0589 3 WRITE(LU,110)
0590 READ(LU,210) IYR
0591 WRITE(LU,111)
0592 READ(LU,210)MCM
0593 WRITE(LU,112)
0594 READ(LU,210) IDAY
0595 WRITE(LU,113)
0596 READ(LU,211) ITIM
0597 WRITE(LU,114)
0598 READ(LU,210) INT
```



ORIGINAL PAGE IS  
OF POOR QUALITY

```
0599 WRITE (LU,131) IYR,MON,IDAY,ITIM,INT
0600 WRITE (LU,189)
0601 READ (LU,269) IANS
0602 IF(IANS.EQ.1111) GO TO 3
0603 IF(IANS.EQ.1111) GO TO 999
0604 GO TO 5
0605 4 IYR = 99
0606 MON = 99
0607 IDAY = 99
0608 ITIM = 9999
0609 INT = 99
0610 C
0611 C SET THE TOWER NUMBER ACCORDING TO THE LAUNCH VEHICLE; IF SHUTTLE,
0612 C TOWER NUMBER = 313, IF TITAN, TOWER NUMBER = 110, IF DELTA, TOWER
0613 C NUMBER = 303.
0614 C
0615 C IF IN THE RESEARCH MODE, ENTER THE TOWER NUMBER
0616 C
0617 5 IF(IRUN.EQ.1) GO TO 23
0618 IF(IVHICL.EQ.0) NTWR = 313
0619 IF(IVHICL.EQ.1) NTWR = 110
0620 IF(IVHICL.EQ.2) NTWR = 303
0621 GO TO 6
0622 23 WRITE (LU,115)
0623 READ (LU,213) NTWR
0624 C
0625 C IF NOT ENTERING ONLY ESSENTIAL DATA, ASK FOR INPUT AT EACH LEVEL.
0626 C
0627 C IF ENTERING ESSENTIAL DATA ONLY, ASK FOR INPUT AT THE 12, 54, AND TDP
0628 C OF THE TOWER LEVELS
0629 C
0630 6 IF(101K.EQ.1) GO TO 8
0631 DO 7 K=1,7
0632 WRITE (LU,130) LEVEL(K)
0633 WRITE (LU,116)
0634 READ (LU,213) IMD(K)
0635 WRITE (LU,117)
0636 READ (LU,210) IUS(K)
0637 WRITE (LU,118)
0638 READ (LU,210) IGS(K)
0639 7 CONTINUE
0640 GO TO 16
0641 C
0642 C IF ENTERING ESSENTIAL DATA ONLY, BEGIN BY GETTING DATA FOR THE
0643 C 12 AND 54 FOOT LEVEL REGARDLESS OF THE LAUNCH VEHICLE.
0644 C SET THE WIND GUST SPEEDS TO 99.
0645 C
0646 8 WRITE (LU,143) NTWR
0647 DO 9 J=1,2
0648 WRITE (LU,130) LEVEL(J)
0649 WRITE (LU,116)
0650 READ (LU,213) IUD(J)
0651 WRITE (LU,117)
0652 READ (LU,210) IUS(J)
0653 IGS(J) = 99
0654 9 CONTINUE
0655 C
0656 C IF ENTERING ONLY ESSENTIAL DATA SET ALL DATA AT 162, 204, 295,
0657 C 394, AND 492 FOOT LEVELS TO 99 OR 999.
0658 C
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0659 DU 11 J-3.7
0660 IWS(J) = 999
0661 IWS(J) = 99
0662 IGS(J) = 99
0663
0664 11 CONTINUE
0665 C
0666 C IF ENTERING ESSENTIAL DATA ONLY AND IF THE LAUNCH VEHICLE IS A
0667 C SHUTTLE CRAFT, THEN PICK UP THE TOP OF THE TOWER DATA (IE,492 FT.)
0668 C
0669 C IF THE LAUNCH VEHICLE IS TITAN, THEN PICK UP THE TOP OF THE TOWER
0670 C DATA (IE, 204 FOOT LEVEL)
0671 IF(IWHICL.EQ.0) GO TO 15
0672 IF(IWHICL.EQ.1) GO TO 13
0673 GO TO 16
0674 13 WRITE (LU,130) LEVEL(4)
0675 WRITE (LU,116)
0676 READ (LU,213) IWS(4)
0677 WRITE (LU,117)
0678 READ (LU,210) IWS(4)
0679 GO TO 16
0680 15 WRITE (LU,130) LEVEL(7)
0681 WRITE (LU,116)
0682 READ (LU,213) IWS(7)
0683 WRITE (LU,117)
0684 READ (LU,210) IWS(7)
0685 C
0686 C WRITE THE VALUES ENTERED BACK TO THE CRT FOR USER APPROVAL BEFORE
0687 C CONTINUING
0688 C
0689 16 IF(IQIK.EQ.0) WRITE (LU,135)HTWR
0690 IF(IQIK.EQ.1) WRITE (LU,140)HTWR
0691 DO 17 K=1,7
0692 IF(IQIK.EQ.1.AND.IWS(K).EQ.999) GO TO 17
0693 IF(IQIK.EQ.1) WRITE (LU,139) LEVEL(K),IWS(K),IWS(K)
0694 IF(IQIK.EQ.0) WRITE (LU,132)LEVEL(K),IWS(K),IWS(K),IGS(K)
0695 17 CONTINUE
0696 WRITE (LU,109)
0697 READ (LU,209) IANS
0698 IF(IANS.EQ.IHS) GO TO 6
0699 IF(IANS.EQ.IHS) GO TO 999
0700 C
0701 C ENTER THE TEMPERATURE AND DEU POINT FOR THE 6 FOOT AMP. 54 FOOT
0702 C LEVELS IF ENTERING ALL DATA, OTHERWISE JUST ENTER THE TEMPERATURES.
0703 C
0704 18 DO 19 K=1,2
0705 WRITE (LU,119) LEVEL2(K)
0706 READ (LU,213) ITEM(K)
0707 IF(IQIK.EQ.1) IOP(K) = 999
0708 IF(IQIK.EQ.1) GO TO 19
0709 WRITE (LU,120) LEVEL2(K)
0710 READ (LU,213) IOP(K)
0711 19 CONTINUE
0712 IF(IQIK.EQ.1) WRITE (LU,141) LEVEL2(1),ITEM(1),LEVEL2(2),ITEM(2)
0713 IF(IQIK.EQ.0) WRITE (LU,133)LEVEL2(1),ITEM(1),IOP(1),LEVEL2(2),
0714 ITEM(2),IOP(2)
0715 WRITE (LU,109)
0716 READ (LU,209) IANS
0717 IF(IANS.EQ.IHS) GO TO 18
0718 IF(IANS.EQ.IHS) GO TO 999
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```

0719 C
0720 C IF ENTERING ONLY ESSENTIAL DATA, SET LAPSE RATE.
0721 C 5 PPM DISTANCE, 25 PPM DISTANCE, SIGMA OF WIND AZIMUTH, AND
0722 C RELATIVE HUMIDITY TO 99.9 OP. 99.
0723 C
0724 C ENTER THE LAPSE RATE, 5 PPM DISTANCE, 25 PPM DISTANCE, THE STANDARD
0725 C DEVIATION OF THE WIND AZIMUTH ANGLE, AND THE RELATIVE HUMIDITY AT
0726 C 54 FEET IF ENTERING ALL DATA.
0727 C
0728 C IF<IOIK.EQ.1> GO TO 21
0729 C 20 WRITE<LU,121>
0730 C READ<LU,221> KATEL
0731 C WRITE<LU,122>
0732 C READ<LU,221> PPM5
0733 C WRITE<LU,123>
0734 C READ<LU,221> PPM25
0735 C WRITE<LU,124>
0736 C READ<LU,221> SDEV
0737 C WRITE<LU,125>
0738 C READ<LU,213> IRH
0739 C WRITE<LU,134>RATEL,PPM5,PPM25,SDEV,IRH
0740 C WRITE<LU,109>
0741 C READ<LU,209> IANS
0742 C IF<IANS.EQ.100> GO TO 20
0743 C IF<IANS.EQ.105> GO TO 999
0744 C GO TO 22
0745 C 21 RATEL = 99.9
0746 C PPM5 = 99.9
0747 C PPM25 = 99.9
0748 C SDEV = 99.9
0749 C IRH = 999
0750 C
0751 C WRITE THE DATA TO THE DISC FILE
0752 C
0753 C 22 CALL CODE
0754 C WRITE<NBUF,126> IYR,MON,IDAY,ITIM,INT,HTWR,IUD<1>,IUS<1>,IGS<1>,
0755 C IUD<2>,IGS<2>,IUD<3>,IUS<3>,IGS<3>,IUD<4>,
0756 C IUS<4>,IGS<4>,ITEM<1>,IDP<1>,IDP<2>,RATEL,PPM5,
0757 C PPM25,SDEV,ITEM<2>,IRH
0758 C
0759 C CALL CODE
0760 C CALL WRITE<NDCB,IERR,NBUF,58>
0761 C IF<IERR.LT.0>WRITE<LU,101> IERR
0762 C CALL CODE
0763 C WRITE<NBUF,120> IUD<5>,IUS<5>,IGS<5>,IUD<6>,IGS<6>,IUD<7>,
0764 C IUS<7>,IGS<7>
0765 C CALL CODE
0766 C CALL WRITE<NDCB,IERR,NBUF,58>
0767 C
0768 C SEE IF THERE IS DATA FOR ANOTHER TOWER USING THE SAME DATE AND TIME
0769 C IF ENTERING ALL DATA.--IF IN OPERATIONAL MODE, DO NOT ASK FOR OTHER
0770 C LAUNCH VEHICLES--AND CLOSE THE FILE. IF IN OPERATIONAL MODE, TEST
0771 C FOR ESSENTIAL DATA ONLY OR ALL DATA INPUT.
0772 C
0773 C IF OTHER LAUNCH VEHICLES OR TOWER NUMBERS ARE TO BE INPUT IN THE
0774 C PRODUCTION OR RESEARCH MODE, LOOP BACK TO THE BEGINNING OF THE
0775 C PROGRAM.
0776 C
0777 C IF<IRUN.EQ.2> GO TO 999
0778 C IF<IRUN.EQ.3> WRITE<LU,129>
0779 C IF<IRUN.EQ.1> WRITE<LU,143>

```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0779 READ (LU,209) IAMS
0780 IF(IAMS.EQ.1HY) GO TO 5
0781 C
0782 C CLOSE THE DATA FILE BEING USED
0783 C
0784 999 HBUF(1) = 2HEN
0785 CALL CODE
0786 CALL WRITF(NDCB,IERR,HBUF,1)
0787 IF(IERR.LT.0) WRITE (LU,101)IERR
0788 CALL CLOSE(NDCB,IERR)
0789 IF(IERR.LT.0) WRITE (LU,101) IERR
0790 C
0791 C TELL THE USER HIS DATA FILE NAME AGAIN
0792 C
0793 WRITE (LU,127) NAM2
0794 C
0795 C
0796 DO 25 K=1,3
0797 NDAT(K) = NAM2(K)
0798 NDAT = NAM2
0799 C
0800 C TERMINATE TOWER
0801 C
0802 RETURN
0803 END
0804 SUBROUTINE FORTD(NAM2,IALT,KNUM)
0805 C++
0806 C++ THIS PROGRAM READS WIND SYSTEM TOWER DATA IN INX- ++
0807 C++ 73337 FORMAT AND REFORMATS THE DATA TO CONFORM TO THE++
0808 C++ SOUNDING/FORECAST DATA FORMAT. ++
0809 C++
0810 C++
0811 C++
0812 C
0813 C
0814 C >>> COMMON AREAS
0815 C
0816 C
0817 C >>> MATH PARAMETERS
0818 COMMON P1,PIOVK2,F143,TUOPI,SOR2PI,DEGRAD,RADGEG,IV2
0819 C
0820 C >>> VEHICLE PARAMETERS
0821 COMMON VPAR(18),SIGICL
0822 C
0823 C >>> OPTION PARAMETERS
0824 COMMON YPOS,YES,ISINFO,IVHICL,IRUN,ICALC,IPLACE,ITIMEZ,IGOIMP,
0825 NUMRUM,LU,NUM,MOHLY,MFORM,LUFANT,ITDU,IPL1,IPL2,IPL3
0826 INTEGER YES
0827 C
0828 C >>> MET DATA
0829 COMMON ALT(30),IDIR(30),SPEED(30),TEMP(30),PRESS(30),FTEMP(30),
0830 SURPEN,FILE(3)
0831 INTEGER FILE
0832 C
0833 C >>> CALCULATION TIME
0834 COMMON ITIME,IDAY,IMONTH(2),IYEAR
0835 C
0836 C >>> SOUNDING TIME
0837 COMMON TSTIME,TGDAY,ISMJNK(2),TSPFAR
0838 C
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0639 C >>> LAMKIN TIME
0640 COMMON LTIME,LTIM,LOAY,LMONTH(2),LYEAR,LAUNTC(10),
0641 LOGICAL LTIME, LVERSH
0642 C
0643 C >>> CLOUD STABILIZATION AND LAYER PARAMETERS
0644 COMMON STBLT,STBHT,STBRZD,STBRHG,STBTIM,CLOUDP,RISTIM(30),BOTLAY,
0645 TOFLAY,CALHT,LAYTOP,LAYBOT,REFLEC,IPTZ
0646 C
0647 C >>> DIFFUSION COEFFICIENTS
0648 COMMON SIGX,SIGZ,SIGZ0,SIGAP,XDIST,YDIST,EXPZ,SOSIGZ,SIGY,SIGAL,
0649 CLDLG,AVSPD,AVDIR,SIGAT
0650 C
0651 C >>> CONCENTRATION AND DOSAGE VALUES AND RANGES
0652 COMMON CONMAX,CONPK,DOSMAX,DOSPK,RHGSMC,RHGSMD
0653 C
0654 C >>> EQUIVALENCE STATEMENT FOR VEHICLE PARAMETERS
0655 EQUIVALENCE (QC1,VPAR(1)),(QC2,VPAR(2)),(QC3,VPAR(3)),
0656 (QT1,VPAR(4)),(QT2,VPAR(5)),(QT3,VPAR(6)),
0657 (CA,VPAR(7)),(CB,VPAR(8)),(CC,VPAR(9)),
0658 (HEATN,VPAR(10)),(HEATA,VPAR(11)),(HEATA,VPAR(12)),
0659 (PHCL,VPAR(13)),(PCO,VPAR(14)),(PCO2,VPAR(15)),
0660 (PAL203,VPAR(16)),(PHO,VPAR(17)),(GAMMAX,VPAR(18))
0661 C
0662 C DIMENSION AND DATA STATEMENTS
0663 C
0664 DIMENSION NAM2(3),NOCB(144),NBUF(58),IUD(30),IMS(30),
0665 ITEM(15),LEVEL(7),NAM3(3),KDCB(144),KBUF(58)
0666 DIMENSION IALT(30),OPTEMP(30),NAME(3)
0667 DATA LEVEL/12,54,162,204,295,394,492/
0668 DATA NAME/2H-R,2HEE,2HD2/
0669 DATA NAM3/ZHRE,2HAD,2HER/
0670 C
0671 C FORMAT STATEMENTS
0672 C
0673 100 FORMAT("OPEN ERROR",I4)
0674 101 FORMAT(" IERR =",I4)
0675 102 FORMAT("CLOSE ERROR",I4)
0676 106 FORMAT('S8A2')
0677 112 FORMAT("KERR =",I3)
0678 144 FORMAT(36X,3(I3,2X,I2,5X,))
0679 145 FORMAT("TUR NBR",I4,I3,I4)
0680 146 FORMAT(I6,IXI3,IXI3,F6.1)
0681 147 FORMAT(I6,IXI3,IXI3)
0682 203 FORMAT(A2)
0683 214 FORMAT(20X,I3,IX,4(I3,2X,I2,5X),I3,32X,I3)
0684 215 FORMAT(I6,IXI3,IXF4.1,F6.1)
0685 216 FORMAT(I6,IXI3,IXF4.1)
0686 248 FORMAT(3A2)
0687 C
0688 C SET THE TOWER NUMBER ACCORDING TO THE VEHICLE NUMBER PASSED
0689 C TO THE SUBROUTINE
0690 C
0691 NTUR = J13
0692 IF (IVHICLEQ.1) NTUR = 110
0693 IF (IVHICLEQ.2) NTUR = 303
0694 C
0695 C OPEN THE FILE SPECIFIED
0696 C
0697 CALL OPEN (MICE,IEPP,NAS,,0)
0698 IF (IERR.LT.0) WRITE (LU,100) IERR
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0037 C
0000 C CREATE THE DISC FILE "REMER"
0001 C
0002 CALL OFENKDCB, YEPR, NAM3, 6)
0003 IF<YEPR,LT.6) CALL CREATKDCB, YEPR, NAM3, 4, 4, 6, 6, 144)
0004 IF<YEPR,LT.6) WRITE<LU,101) YEPR
0005 C
0006 C READ LINES OF THE DATA FILE UNTIL DATA IS FOUND
0007 C
0008 3 CALL READF(NDCB, IERR, NBUF, 50, LEN)
0009 IF<IERR,LT.6) WRITE<LU,101) IERR
0010 IF<NBUF<1).EQ.2) GO TO 4
0011 IF<NBUF<1).EQ.2) GO TO 3
0012 GO TO 3
0013 C
0014 C READ THE DATA HEADINGS
0015 C
0016 4 CONTINUE
0017 CALL CODE
0018 READ<NBUF,106) (NBUF<1), I=1, LEN)
0019 CALL READF(NDCB, IERR, NBUF, 50, LEN)
0020 IF<IERR,LT.6) WRITE<LU,101) IERR
0021 IF<NBUF<1).EQ.2) GO TO 5
0022 CALL CODE
0023 READ<NBUF,106) (NBUF<1), I=1, LEN)
0024 CALL CODE
0025 CALL READF(NDCB, IERR, NBUF, 50, LEN)
0026 IF<IERR,LT.6) WRITE<LU,101) IERR
0027 IF<NBUF<1).EQ.2) GO TO 5
0028 CALL CODE
0029 READ<NBUF,106) (NBUF<1), I=1, LEN)
0030 C
0031 C READ THE DATA FROM THE TMX 7337 FORMATTED FILE AND WRITE IT
0032 C TO THE DISC FILE "REMER" IN THE SAME FORMAT AS THE SOUNDING/
0033 C FORECAST
0034 C
0035 DO 7 K=1,22
0036 CALL READF(NDCB, IERR, NBUF, 50, LEN)
0037 IF<IERR,LT.6) WRITE<LU,101) IERR
0038 IF<NBUF<1).EQ.2) GO TO 99
0039 CALL CODE
0040 READ<NBUF,214) (TUR, IWD<1), IUS<1), IUD<2), IUS<2), IUD<3), IUS<3),
0041 , IWD<4), IUS<4), ITEM<1), ITEM<2)
0042 CALL CODE
0043 IF<TUR,NE,NTUR) GO TO 7
0044 CALL READF(NDCB, IERR, NBUF, 50, LEN)
0045 IF<NBUF<1).EQ.2) GO TO 99
0046 IF<IERR,LT.6) WRITE<LU,101) IERR
0047 CALL CODE
0048 READ<NBUF,144) IUD<5), IUS<5), IUD<6), IUS<6), IUD<7), IUS<7)
0049 CALL CODE
0050 WRITE<NBUF,145) FOUR
0051 CALL CODE
0052 CALL WRITE(KDCB, YEPR, NBUF, 6)
0053 CALL CODE
0054 WRITE<NBUF,146) LEVEL<1), IUD<1), IUS<1), ITEM<1)
0055 CALL CODE
0056 CALL WRITE(KDCB, YEPR, NBUF, 10)
0057 IF<YEPR<1) WRITE<LU,101) YEPR
0058 . I C C . .
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0959 WRITE (KBUF,146) LEVEL(2),IUD(2),IUS(2),ITEM(2)
0960 CALL CODE
0961 CALL WRITF(KDCB,KERR,KBUF,10)
0962 IF(KERR.LT.0) WRITE (LU,101) KERR
0963 IF(ITUR.EQ.303) GO TO 99
0964 I = 7
0965 IF(ITUR.EQ.110) I = 4
0966 CALL CODE
0967 WRITE (KBUF,147) LEVEL(1),IUD(1),IUS(1)
0968 CALL CODE
0969 CALL WRITF(KDCB,KERR,KBUF,10)
0970 IF(KERR.LT.0) WRITE (LU,101) KERR
0971 7 CONTINUE
0972 C
0973 C PUT AN END OF FILE MARK ON THE DATA FILE "READER" AND
0974 C CLOSE BOTH FILES.
0975 C
0976 99 NBUF(1) = ZHEN
0977 CALL CODE
0978 WRITE (KBUF,203) NBUF(1)
0979 CALL CODE
0980 CALL WRITF(KDCB,KERR,KBUF,1)
0981 CALL CLOSE(NDCB,IERR)
0982 CALL CLOSE(KDCB,KERR)
0983 IF(IERR.LT.0) WRITE (LU,102) IERR
0984 IF(KERR.LT.0) WRITE (LU,102) KERR
0985 CALL OPEN(KDCB,KERR,NAM3,0)
0986 IF(KERR.LT.0) WRITE (LU,112) KERR
0987 C
0988 C LOOP UNTIL THE CORRECT TOWER NUMBER IS FOUND
0989 C
0990 J3 CALL READF(KDCB,KERR,KBUF,40,LEN)
0991 IF (KERR.LT.0) WRITE (LU,112) KERR
0992 IF (KBUF(1).EQ.2)HEN) GO TO 999
0993 C
0994 C READ IN THE DATA
0995 C
0996 NUM1 = INUM + 1
0997 NUM2 = INUM + 2
0998 NUM3 = INUM + 3
0999 DO 5 J=NUM1,NUM2
1000 CALL READF(KDCB,KERR,KBUF,40,LEN)
1001 IF (KERR.LT.0) WRITE (LU,112) KERR
1002 IF (KBUF(1).EQ.2)HEN) GO TO 999
1003 CALL CODE
1004 READ (KBUF,215) IALT(J),IDIR(J),SPEED(J),TEMP(J)
1005 C
1006 C CONVERT THE TEMPERATURE IN THE TOWER DATA FROM DEGREES
1007 C F TO DEGREES C.
1008 C
1009 TEMP(J) = .555555*(TEMP(J)-32.0)
1010 5 CONTINUE
1011 C
1012 C IF THE LAUNCH VEHICLE IS A DELTA (TOWER NUMBER 303),
1013 C THERE IS NO OTHER DATA
1014 C
1015 IF(IVHICL.EQ.2) GO TO 999
1016 CALL READF(KDCB,KERR,KBUF,40,LEN)
1017 IF (KERR.LT.0) WRITE (LU,1015) KERR
1018 IF (KBUF(1).EQ.2)HEN) GO TO 999
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
1019 CALL CODE
1020 READ (KBUF,216) IALT(NUM3),IDTR(NUM3),SPEED(NUM3)
1021 C
1022 C CLOSE THE DATA FILE "READER"
1023 C
1024 999 CALL CLOSE(KDCB,KERR)
1025 IF (KERR.LT.0) WRITE (LU,101) KERR
1026 IF (IWHICL.EQ.2) KNUM = NUM + 2
1027 IF (IWHICL.NE.2) KNUM = NUM + 3
1028 DO 95 J=NUM1,KNUM
1029 OPTMP(J) = 0.0
1030 95 PRESS(J) = 0.0
1031 C
1032 C TERMINATE FORTD
1033 C
1034 RETURN
1035 END
1036 ENDS
```





```

0059 C
0060 C*** INPUT FORMATS
0061 90 FURMAT(1)
0062 C
0063 C OUTPUT FORMAT STATEMENTS
0064 C
0065 100 FORMAT (F6.1)
0066 101 FORMAT (F7.2)
0067 102 FORMAT (F4.1)
0068 200 FORMAT ('1-27X EXHAUST CLOUD'/'0LEVEL'4X'ALITUDE'17X
0069 '(SECONDS)'4X'(METERS)'4X'(DEGREES)')
0070 201 FORMAT (2X13.5XF7.1,5X'ADIABATIC'5X3A2,6XF7.1,7XF5.1)
0071 202 FORMAT (2X13.5XF7.1,6X'STABLE'7X3A2,6XF7.1,7XF5.1)
0072 203 FORMAT ('/'0***CLOUD STABILIZATION***'/'
0073 6X'HEIGHT(M)'/'F6.1/'
0074 6X'Stabilization TIME AFTER LAUNCH(SEC)'/'F5.1/'
0075 6X'RANGE FROM PAD(M)'/'F7.1/'
0076 6X'DIRECTION FROM PAD( DEG)'/'F5.1)
0077
0078 204 FORMAT (F6.1)
0079 205 FORMAT ('/'0***TOP OF SURFACE LAYER METEOROLOGICAL PARAMETERS'
0080 '****'/'
0081 6X'HEIGHT(M)'/'F6.1/'
0082 6X'WIND DIRECTION(DEG)'/'13/'
0083 6X'WIND SPEED(M/SEC)'/'F4.1)
0084 206 FORMAT ('/'0***DIFFUSION PARAMETERS***'/'
0085 6X'MEAN SPEED(M/SEC)'/'F4.1/'
0086 6X'MEAN TRANSPORT DIRECTION(DEG)'/'F5.1)
0087 207 FORMAT (F3.0)
0088 208 FORMAT ('/'0SIGMA OF WIND AZIMUTH ANGLE, SIGAZ)'/'F4.1)
0089 209 FORMAT ('/'0EFFECTIVE CLOUD HEIGHT(M)'/'F6.1)
0090 C
0091 C TYPE AND DIMENSION STATEMENTS
0092 C
0093 INTEGER RMETP(3),RCONC(3)
0094 REAL LEAST
0095 DIMENSION NAME(3),NAMEF(3),ILINE(32),IDATAF(10),IERS(32)
0096 C
0097 C EQUIVALENCE STATEMENT
0098 C
0099 EQUIVALENCE (ILIN23,ILINE(23)),(ILIN24,ILINE(24))
0100 C
0101 C DATA STATEMENTS
0102 C
0103 DATA NAME/2H=R.2HEE,2HD2/, NAMEF/2H?R,2HCL,2HDS/
0104 DATA RMETP/2HRM,2HTP,1HS/
0105 DATA RCONC/2HRC,2HOM,1HS/
0106 DATA IERS/32*2H /
0107 C
0108 C CALL SUBROUTINE GRAF TO INITIALIZE PLASMASCOPE GRAPHIC MODE
0109 C
0110 CALL GRAF(1)
0111 CALL DREAD(NAMEF,2,ILINE)
0112 CALL DREAD(NAMEF,2,ILINE)
0113 C
0114 C READ THE COMMON DATA FILE
0115 C
0116 CALL RWDIS(NAME,0)
0117 C*** END CALL TO READ TO TEST OPEN
0118 C*** I. Ph... ,Math... ,IL...

```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0119 C INITIALIZE SOME LOCAL VARIABLES
0120 C
0121 C
0122 C RSTINC ) - CLOUD RISE TIME
0123 C IAS - 0 = ADIABATIC
0124 C I = STABLE
0125 C ALTINC - ALTITUDE INCREMENT
0126 C ITERAT - ITERATION COUNTER
0127 C
0128 C ANGY = 0.0
0129 C RNCX = 0.0
0130 C RSTINC(1) = 0.0
0131 C ALTINC = 0.0
0132 C STBRNG = 9.0
0133 C STBAZD = 0.0
0134 C
0135 C*** INITIALIZE DISPLAY LU FLAG
0136 C CALL VERSNK(1)
0137 C LVERSH = 1 .EQ. 0
0138 C
0139 C WRITE OUT THE EXHAUST CLOUD HEADER
0140 C
0141 C WRITE (LUPRINT,200)
0142 C
0143 C CALCULATE SOME QUANTITIES TO BE USED IN SUBSEQUENT DO LOOP
0144 C
0145 C ALPHAC = 5.12913086E-02*(TEMP(1) + 273.16)**SURDEH*GANMAX**3
0146 C ALPHAC = ALPHAC/(HEATN * QC1)
0147 C GT = 9.0/(TEMP(1) + 273.16)
0148 C
0149 C DO LOOP TO CALCULATE EXHAUST CLOUD PARAMETERS
0150 C
0151 C C C***THE FOLLOWING CHANGES ARE AN ATTEMPT TO USE THE ALGORITHM
0152 C C C***WRITTEN IN CR-2631, PAGE 15, FOR A LEAST SQUARE REGRESSION
0153 C C C***FIT FOR THE HEIGHT RATE OF CHANGE FOR POTENTIAL TEMPERATURE.
0154 C C C***...JSH
0155 C IFLAG = 1
0156 C ZSUM = 0.0
0157 C PTSUM = 0.0
0158 C S1 = 9.0
0159 C S2 = 0.0
0160 C
0161 C DO 9 I=2,NUM
0162 C IMI = I - 1
0163 C IAS = I
0164 C POT = PTEMP(I) -273.16
0165 C
0166 C CALCULATE SLOPE OF POTENTIAL TEMPERATURE, SPEED, AND
0167 C DIRECTION IN LAYER
0168 C
0169 C DALT = ALT(I) - ALT(IMI)
0170 C OPTEMP = (PTEMP(I) - PTEMP(IMI))/DALT
0171 C CSPEED = (SPEED(I) - SPEED(IMI))/DALT
0172 C DDEG = ABS(FLOAT(DIR(I) - DIR(IMI)))
0173 C IS<DDEG.GT.180.) DDEG = 360.0 - DDEG
0174 C DIR = DDEG/DALT
0175 C IF(IPTZ.EQ.1)GO TO 57
0176 C
0177 C CALCULATE METEOROLOGICAL OBSERVATIONS
0178 C
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```

0179 1 IF(ALTINC.EQ.0.0)GO TO 44
0180 S1 = S1 - S1INC
0181 S2 = S2 - S2INC
0182 ZSUM = ZSUM - Z
0183 PTBUM = PTBUM - PT
0184 Z = ALT(I) - ALTINC
0185 TERP = (ALT(I) - ALTINC - ALT(IMI))/ DALT
0186 PT = POT*TERP
0187 GO TO 55
0188 44 Z = ALT(I)
0189 PT = POT
0190 55 ZSUM = ZSUM + Z
0191 PTBUM = PTBUM + PT
0192 ZAV = ZSUM/FLOAT(I)
0193 PTAV = PTBUM/FLOAT(I)
0194 S1INC = (Z - ZAV)*(PT - PTAV)
0195 S2INC = (Z - ZAV)**2
0196 S1 = S1 + S1INC
0197 S2 = S2 + S2INC
0198 STAB = GT*(S1/S2)
0199 WRITE(6,1232)I,STAB,ALTINC,Z,S1,PT,PTAV
0200 1232 FORMAT(IX,'I,STAB,ALTINC,Z,S1,PT,PTAV=',IX,13,6(1X,E10.8))
0201 57 ZALPHA = ALT(I) - ALT(I) - ALTINC
0202 ALPHA = ALPHA + ZALPHA**4/(AA + ZALPHA**BB + CC)
0203 C
0204 C CALCULATE POTENTIAL TEMPERATURE FACTOR
0205 C
0206 IF(IPTZ.EQ.1)STAB = GT * (PTMP(I) - ALTINC * GPTE1P - PTMP(I))/
0207 (ALT(I) - ALTINC - ALT(I) + 1.0E-7)
0208 C
0209 C CALCULATION FOR ADIABATIC RISE
0210 C
0211 IF(STAB .GT. 0.000001)GO TO 2
0212 R1STINC(I) = SQRT(ALPHA)
0213 IAS = 0
0214 GO TO 5
0215 C
0216 C CALCULATION FOR STABLE CLOUD RISE
0217 C
0218 C C2 = ARGUMENT OF ARC COSINE (MUST BE LESS THAN -1)
0219 C
0220 2 C2 = 1.0 - 0.5 * ALPHA * STAB
0221 WRITE(6,122)I,ALPHA,STAB,C2
0222 122 FORMAT(IX,'I,ALPHA,STAB,C2 =',IX,13,3(1X,E14.8))
0223 IF(C2 .LT. -1.0)GO TO 4
0224 C3 = C2/SQRT(1.0 - C2 * C2)
0225 R1STINC(I) = (PI*VR2 - ATAN(C3))/SQRT(STAB)
0226 IF(IFLAG.EQ.0) GO TO 11
0227 GO TO 5
0228 C
0229 C ITERATE IN LAYER
0230 C
0231 4 ALTINC = ALTINC + 0.1
0232 IFLAG = 0
0233 IF(IPTZ.EQ.1)GO TO 57
0234 GO TO 1
0235 C
0236 C CALCULATE RANGE AND DIRECTION
0237 C
238

```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0239          (RISTIM(IMI) - RISTIM(I))
0240
0241 C*** AVERAGE WIND DIRECTION...BE CAREFUL TO ADJUST FOR PROPER VECTOR DIRECTION
0242 C
0243       DELDIR = DEGRAD * (.5*(FLOAT(IDIR(I)) + IDIR(IMI))) -
0244         180.*(ABS(IDIR(I)) - IDIR(IMI))/180)
0245       DIRECT = RADDEC * DELDIR
0246 C
0247 C*** FIND ACTUAL WIND TRANSPORT DIRECTION...ADD PI FOR DELDIR <180...ELSE SUB
0248 C
0249       IFLIP = 1
0250       IFCDELDIR .GT. PI) IFLIP = -1
0251       DELDIR = DELDIR + IFLIP*PI
0252       RNGY = RNGY + STBRNG * SIN(PI - DELDIR)
0253       RNGX = RNGX + STBRNG * COS(PI - DELDIR)
0254       STBAZD = ATAN2(RNGY,RNGX)
0255       STBAZD = RADDEC * (PI - STBAZD)
0256       STBRNG = SORT(RNGY * RNGY + RNGX * RNGX)
0257 C
0258 C   WRITE OUT THE VARIABLES WITH THE APPROPRIATE FORMAT STATEMENT
0259 C   BASED OF WHETHER OR NOT CLOUD IS ADIABATIC OR STABLE
0260 C
0261       CALL BLANK(IDATAF,3)
0262       IFCALTING .GT. 0.0XGO TO 6
0263       CALL CODE
0264       WRITE (IDATAF,100) RISTIM(I)
0265       6 IFC(LAS .NE. 0)GO TO 7
0266       WRITE (LUPRNT,201) I,ALT(I),(IDATAF(J),J=1,3),STBRNG,STBAZD
0267       GO TO 8
0268       7 WRITE (LUPRNT,202) I,ALT(I),(IDATAF(J),J=1,3),STBRNG,STBAZD
0269       8 IFCALTING .NE. 0.0XGO TO 11
0270       9 CONTINUE
0271 C
0272 C   CALCULATE AND WRITE OUT STABILIZATION HEIGHT AND TIME
0273 C
0274       11 STBRNG = 0.5 * (SPEED(IMI) - ALTIM * GSPEED + SPEED(I)) *
0275         (RISTIM(I) - RISTIM(IMI))
0276       DALI = 0.0072665 * (FLOAT(IDIR(I)) + IDIR(IMI)) - GDIR * ALTIMC)
0277       RNGY = RNGY - STBRNG * SIN(PI - DALI)
0278       RNGX = RNGX - STBRNG * COS(PI - DALI)
0279       STBAZD = ATAN2(RNGY,RNGX)
0280       STBAZD = RADDEC * (PI - STBAZD)
0281       STBRNG = SORT(RNGY * RNGY + RNGX * RNGX)
0282       STBALI = ALT(I) - ALTIMC
0283       WRITE (LUPRNT,203) STBALI,RISTIM(I),STBRNG,STBAZD
0284       TIST = PI/(SORT(STAB))
0285 C
0286 C   STORE THE INDEX OF THE ESTIMATED TOP OF THE SURFACE LAYER
0287 C
0288       LAYTOP = I
0289 C
0290 C   LOAD THE CLOUD RISE TIME ARRAY
0291 C
0292       STBTIM = RISTIM(LAYTOP)
0293       I = I + 1
0294       GO 12 J-I,IMIN
0295       12 RISTIM(J) = STBTIM
0296 C
0297 C   CALL FTOP TO COMPUTE THE INDEX OF THE TOP OF THE SURFACE LAYER
0298 C
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0299 C CALL FTOP
0300 C
0301 C IS THIS A RESEARCH, OPERATIONAL, OR PRODUCTION RUN?
0302 C
0303 C IF<IRUN .LE. 2>GO TO 18
0304 C
0305 C PRODUCTION RUN -- IF TOPLAY IS UNDEFINED, USE LAYTOP AS ESTIMATED
0306 C
0307 C 15 IF<TOPLAY .LE. 0.0>GO TO 26
0308 C
0309 C CALCULATE LAYTOP BASED ON VALUE OF TOPLAY
0310 C
0311 C LEASTD = 9999999.9
0312 C DO 16 I=1,MUM
0313 C DIFF = ABS<ALT(I) - TOPLAY>
0314 C IF<DIFF .GT. LEASTD>GO TO 16
0315 C LEASTD = DIFF
0316 C LAYTOP = I
0317 C 16 CONTINUE
0318 C GO TO 26
0319 C
0320 C CALL THE SUBROUTINE RSGAZ TO CALCULATE THE VALUE OF SIGAZ
0321 C NOTE THAT THE FLAG ITDU IS USED TO BRANCH WHEN TOWER DATA
0322 C IS BEING USED.
0323 C
0324 C 18 J1 = 1
0325 C J2 = 0
0326 C J3 = 0
0327 C IF<ITDU.EQ.0> GO TO 17
0328 C J1 = 1
0329 C J2 = 3
0330 C J3 = 4
0331 C GO TO 25
0332 C 17 DO 19 J=1,MUM
0333 C IF<ABS<ALT(J)-304.0> .LE. 1.0>J3 = J
0334 C
0335 C **THE TEST FOR DETERMINING J3 AND J2 HAS BEEN ALTERED SINCE 1.0
0336 C **AND 1000.0 ARE TOO RIGID FOR WORKABLE STANDARDS.
0337 C
0338 C IF<ABS<ALT(J)-304.0> .LT. ABS<ALT(J3)-304.0>>J3 = J
0339 C IF<ALT(J).LE.304.8 .AND. ALT<J+1>.GT.304.8> J3 = J
0340 C IF<ABS<PRESS<J>-1000.0>.LT. ABS<PRESS<J2>-1000.0>>J2 = J
0341 C 19 CONTINUE
0342 C SIGAZ = 7.0
0343 C 25 IF<J2.NE.0 .AND. J3.NE.0>CALL RSGAZ<J1,J2,J3,SIGAZ>
0344 C WRITE<6,787> SIGAZ,NAMEF,LAYTOP
0345 C 787 FORMAT<' RSGAZ IN RCLD8 IS: ',F0.4,' FILE: ',3A2,' LAYTOP: ',I4>
0346 C
0347 C WRITE OUT THE ESTIMATED TOP OF SURFACE LAYER -- READ IN
0348 C THE ONE TO BE USED -- CALCULATE LAYTOP
0349 C
0350 C CALL DREAD<NAMEF,2,ILINE>
0351 C CALL LERS<YPOS,LVERSI>
0352 C CALL CHAR<0.0,YPOS,0,ILINE,64,0.0>
0353 C CALL CODE
0354 C WRITE<IDATAF,204> ALT<LAYTOP>
0355 C TOPLAY = ALT<LAYTOP>
0356 C CALL CHAR<320.0,YPOS,0,IDATAF,6,0.0>
0357 C YPOS = YPOS - 32.0
0358 C
0359 C
0360 C
0361 C
0362 C
0363 C
0364 C
0365 C
0366 C
0367 C
0368 C
0369 C
0370 C
0371 C
0372 C
0373 C
0374 C
0375 C
0376 C
0377 C
0378 C
0379 C
0380 C
0381 C
0382 C
0383 C
0384 C
0385 C
0386 C
0387 C
0388 C
0389 C
0390 C
0391 C
0392 C
0393 C
0394 C
0395 C
0396 C
0397 C
0398 C
0399 C
0400 C
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0359 IF(IROWN .GE. 2)GO TO 24
0360 CALL DREAD(NAMEF,3,ILINE)
0361 CALL IERS(YPOS,LVERSN)
0362 CALL CHAR(0,0,YPOS,0,ILINE,6,3,0)
0363 CALL CHAR(56,0,YPOS,0,IERS,1,3,0)
0364 CALL CHAR(64,0,YPOS,0,ILINE(5),9,3,0)
0365 CALL CHAR(160,0,YPOS,0,ILINE(11),44,0,0)
0366 NIN = 6
0367 CALL BLANK(IDATAF,3)
0368 IX = 0
0369 TOPIX = IX
0370 IF(.NOT.LVERSN) GO TO 20
0371 CALL IK2,JTYPE,463,0,0,DATAF,NIN,0,31,0,31,IX,IY)
0372 IF(JTYPE .EQ. 1)GO TO 22
0373 GO TO 21
0374 CONTINUE
0375 READ(1,10)TOPIN
0376 IF(TOPIN.EQ.110)STOP
0377 IF(TOPIN .LE. TOPIX) GO TO 22
0378 TOPLAY = TOPIN
0379 IX = 20
0380 CALL CHAR(0,0,YPOS,0,ILINE,10,0,0)
0381 CALL CHAR(47,0,YPOS,0,IERS,40,0,0)
0382 YPOS = YPOS - 32
0383 GO TO 22
0384 CONTINUE
0385 CALL CHAR(0,0,YPOS,0,ILINE,6,0,0)
0386 CALL CHAR(47,0,YPOS,0,IERS,40,0,0)
0387 YPOS = YPOS - 32.0
0388 CALL CODE
0389 READ (IDATAF,101) TOPLAY
0390 GO TO 15
0391 22 IF(IX .GT. 9)GO TO 23
0392 CALL CHAR(0,0,YPOS,0,ILINE,10,0,0)
0393 CALL CHAR(47,0,YPOS,0,IERS,46,0,0)
0394 YPOS = YPOS - 32.0
0395 GO TO 15
0396 23 CALL CHAR(0,0,YPOS,0,ILINE,6,0,0)
0397 CALL CHAR(56,0,YPOS,0,IERS,10,0,0)
0398 CALL CHAR(360,0,YPOS,0,IERS,10,0,0)
0399 YPOS = YPOS - 32.0
0400 C
0401 C
0402 C
0403 C
0404 C
0405 C
0406 C
0407 C
0408 C
0409 C
0410 C
0411 C
0412 C
0413 C
0414 C*** WRITE TO THE PUNCH IS TEMPORARILY DISABLED
0415 C
0416 C
0417 C
0418 C*** IF(MOPLY .EQ. 0)STOP
0419 C
0420 C
0421 C
0422 C
0423 C
0424 C
0425 C
0426 C
0427 C
0428 C
0429 C
0430 C
0431 C
0432 C
0433 C
0434 C
0435 C
0436 C
0437 C
0438 C
0439 C
0440 C
0441 C
0442 C
0443 C
0444 C
0445 C
0446 C
0447 C
0448 C
0449 C
0450 C
0451 C
0452 C
0453 C
0454 C
0455 C
0456 C
0457 C
0458 C
0459 C
0460 C
0461 C
0462 C
0463 C
0464 C
0465 C
0466 C
0467 C
0468 C
0469 C
0470 C
0471 C
0472 C
0473 C
0474 C
0475 C
0476 C
0477 C
0478 C
0479 C
0480 C
0481 C
0482 C
0483 C
0484 C
0485 C
0486 C
0487 C
0488 C
0489 C
0490 C
0491 C
0492 C
0493 C
0494 C
0495 C
0496 C
0497 C
0498 C
0499 C
0500 C
0501 C
0502 C
0503 C
0504 C
0505 C
0506 C
0507 C
0508 C
0509 C
0510 C
0511 C
0512 C
0513 C
0514 C
0515 C
0516 C
0517 C
0518 C
0519 C
0520 C
0521 C
0522 C
0523 C
0524 C
0525 C
0526 C
0527 C
0528 C
0529 C
0530 C
0531 C
0532 C
0533 C
0534 C
0535 C
0536 C
0537 C
0538 C
0539 C
0540 C
0541 C
0542 C
0543 C
0544 C
0545 C
0546 C
0547 C
0548 C
0549 C
0550 C
0551 C
0552 C
0553 C
0554 C
0555 C
0556 C
0557 C
0558 C
0559 C
0560 C
0561 C
0562 C
0563 C
0564 C
0565 C
0566 C
0567 C
0568 C
0569 C
0570 C
0571 C
0572 C
0573 C
0574 C
0575 C
0576 C
0577 C
0578 C
0579 C
0580 C
0581 C
0582 C
0583 C
0584 C
0585 C
0586 C
0587 C
0588 C
0589 C
0590 C
0591 C
0592 C
0593 C
0594 C
0595 C
0596 C
0597 C
0598 C
0599 C
0600 C
0601 C
0602 C
0603 C
0604 C
0605 C
0606 C
0607 C
0608 C
0609 C
0610 C
0611 C
0612 C
0613 C
0614 C
0615 C
0616 C
0617 C
0618 C
0619 C
0620 C
0621 C
0622 C
0623 C
0624 C
0625 C
0626 C
0627 C
0628 C
0629 C
0630 C
0631 C
0632 C
0633 C
0634 C
0635 C
0636 C
0637 C
0638 C
0639 C
0640 C
0641 C
0642 C
0643 C
0644 C
0645 C
0646 C
0647 C
0648 C
0649 C
0650 C
0651 C
0652 C
0653 C
0654 C
0655 C
0656 C
0657 C
0658 C
0659 C
0660 C
0661 C
0662 C
0663 C
0664 C
0665 C
0666 C
0667 C
0668 C
0669 C
0670 C
0671 C
0672 C
0673 C
0674 C
0675 C
0676 C
0677 C
0678 C
0679 C
0680 C
0681 C
0682 C
0683 C
0684 C
0685 C
0686 C
0687 C
0688 C
0689 C
0690 C
0691 C
0692 C
0693 C
0694 C
0695 C
0696 C
0697 C
0698 C
0699 C
0700 C
0701 C
0702 C
0703 C
0704 C
0705 C
0706 C
0707 C
0708 C
0709 C
0710 C
0711 C
0712 C
0713 C
0714 C
0715 C
0716 C
0717 C
0718 C
0719 C
0720 C
0721 C
0722 C
0723 C
0724 C
0725 C
0726 C
0727 C
0728 C
0729 C
0730 C
0731 C
0732 C
0733 C
0734 C
0735 C
0736 C
0737 C
0738 C
0739 C
0740 C
0741 C
0742 C
0743 C
0744 C
0745 C
0746 C
0747 C
0748 C
0749 C
0750 C
0751 C
0752 C
0753 C
0754 C
0755 C
0756 C
0757 C
0758 C
0759 C
0760 C
0761 C
0762 C
0763 C
0764 C
0765 C
0766 C
0767 C
0768 C
0769 C
0770 C
0771 C
0772 C
0773 C
0774 C
0775 C
0776 C
0777 C
0778 C
0779 C
0780 C
0781 C
0782 C
0783 C
0784 C
0785 C
0786 C
0787 C
0788 C
0789 C
0790 C
0791 C
0792 C
0793 C
0794 C
0795 C
0796 C
0797 C
0798 C
0799 C
0800 C
0801 C
0802 C
0803 C
0804 C
0805 C
0806 C
0807 C
0808 C
0809 C
0810 C
0811 C
0812 C
0813 C
0814 C
0815 C
0816 C
0817 C
0818 C
0819 C
0820 C
0821 C
0822 C
0823 C
0824 C
0825 C
0826 C
0827 C
0828 C
0829 C
0830 C
0831 C
0832 C
0833 C
0834 C
0835 C
0836 C
0837 C
0838 C
0839 C
0840 C
0841 C
0842 C
0843 C
0844 C
0845 C
0846 C
0847 C
0848 C
0849 C
0850 C
0851 C
0852 C
0853 C
0854 C
0855 C
0856 C
0857 C
0858 C
0859 C
0860 C
0861 C
0862 C
0863 C
0864 C
0865 C
0866 C
0867 C
0868 C
0869 C
0870 C
0871 C
0872 C
0873 C
0874 C
0875 C
0876 C
0877 C
0878 C
0879 C
0880 C
0881 C
0882 C
0883 C
0884 C
0885 C
0886 C
0887 C
0888 C
0889 C
0890 C
0891 C
0892 C
0893 C
0894 C
0895 C
0896 C
0897 C
0898 C
0899 C
0900 C
0901 C
0902 C
0903 C
0904 C
0905 C
0906 C
0907 C
0908 C
0909 C
0910 C
0911 C
0912 C
0913 C
0914 C
0915 C
0916 C
0917 C
0918 C
0919 C
0920 C
0921 C
0922 C
0923 C
0924 C
0925 C
0926 C
0927 C
0928 C
0929 C
0930 C
0931 C
0932 C
0933 C
0934 C
0935 C
0936 C
0937 C
0938 C
0939 C
0940 C
0941 C
0942 C
0943 C
0944 C
0945 C
0946 C
0947 C
0948 C
0949 C
0950 C
0951 C
0952 C
0953 C
0954 C
0955 C
0956 C
0957 C
0958 C
0959 C
0960 C
0961 C
0962 C
0963 C
0964 C
0965 C
0966 C
0967 C
0968 C
0969 C
0970 C
0971 C
0972 C
0973 C
0974 C
0975 C
0976 C
0977 C
0978 C
0979 C
0980 C
0981 C
0982 C
0983 C
0984 C
0985 C
0986 C
0987 C
0988 C
0989 C
0990 C
0991 C
0992 C
0993 C
0994 C
0995 C
0996 C
0997 C
0998 C
0999 C
1000 C
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0419 C          CHANGE THE TOPLAY VALUE?
0420 C
0421 C
0422 C          CALL GRAF(I)
0423 CC         YPOS = 474.0
0424 C          IF(LVERSN) YPOS = 474.0
0425 C          CALL DREAD(NAMEF,5,ILINE)
0426 C          CALL LERS(YPOS,LVERSN)
0427 C          CALL CHAR(0.0,YPOS,6,ILINE,50.0,0)
0428 C          CALL CODE
0429 C          WRITE (IDATAF,101) TOPLAY
0430 C          CALL CHAR(400.0,YPOS,0,IDATAF,7.0,0)
0431 C          ALT(LAYTOP) = TOPLAY
0432 C          YPOS = YPOS - 32.0
0433 C          GO TO 15
0434 C
0435 C          WRITE (LUPRNT,205) TOPLAY, IDIR(LAYTOP), SPEED(LAYTOP)
0436 C          AND SPEED AT THE TOP
0437 C
0438 C          26 CONTINUE
0439 C          WRITE (LUPRNT,205) TOPLAY, IDIR(LAYTOP), SPEED(LAYTOP)
0440 C
0441 C          CALCULATE SOURCE STRENGTH
0442 C
0443 C          SIGNCL = 2.276E3 * PHCL * QCI * A.. * (TEMP(I) + 273.16) /
0444 C             PRES6(I) * TOPLAY**BB
0445 C
0446 C          CALCULATE AND WRITE OUT THE MEAN WIND SPEED, AVSPD, AND
0447 C             DIRECTION, AVDIR
0448 C
0449 C          DO 28 I=2,LAYTOP
0450 C             IF(IABS(IDIR(I)) - IDIR(I-1)) .LT. 100) GO TO 28
0451 C             DO 27 J=1,LAYTOP
0452 C                27 IF(IDIR(J) .LT. 100) IDIR(J) = IDIR(J) + 360
0453 C                GO TO 31
0454 C             28 CONTINUE
0455 C
0456 C             31 AVSPD = 0.0
0457 C             AVDIR = 0.0
0458 C             DO 32 I=2,LAYTOP
0459 C                IMI = I - 1
0460 C                DALT = ALT(I) - ALT(IMI)
0461 C                AVSPD = AVSPD + 0.5 * (SPEED(I) + SPEED(IMI)) * DALT
0462 C                32 AVDIR = AVDIR + 0.5 * FLOAT(IDIR(I) + IDIR(IMI)) * DALT
0463 C
0464 C             DO 34 I=1,LAYTOP
0465 C                34 IF(IDIR(I) .GT. 360) IDIR(I) = IDIR(I) - 360
0466 C
0467 C             DALT = ALT(LAYTOP) - ALT(1)
0468 C             AVSPD = AVSPD/DALT
0469 C             AVDIR = AVDIR/DALT
0470 C             IF(AVDIR .GT. 100.0) GO TO 35
0471 C             AVDIR = AVDIR + 100.0
0472 C             GO TO 36
0473 C             35 AVDIR = AVDIR - 100.0
0474 C
0475 C             36 WRITE (LUPRNT,206) AVSPD,AVDIR
0476 C
0477 C
0478 C          IS THIS A PEESEPCN OPERATIONAL, OR PRODUCTION RUN?
```



ORIGINAL PAGE IS  
OF POOR QUALITY

```
0479 C IF<IRUN .EQ. 3>GO TO 45
0480 C RESEARCH OR OPERATIONAL RUN -- ALLOW THE CALCULATED VALUE OF
0481 C SIGAZ TO BE CHANGED
0482 C
0483 C
0484 C CALL DREAD<NAMEF,6,ILINE>
0485 C CALL LERS<YPOS,LVERSN>
0486 C CALL CODE
0487 C WRITE<ILIN23,102> SIGAZ
0488 C CALL CHAR<0,0,YPOS,0,ILINE,43,3,0>
0489 C CALL CHAR<352,0,YPOS,0,ILINE<23>,4,0,0>
0490 C CALL CHAR<394,0,YPOS,0,ILINE<25>,9,3,0>
0491 C CALL CHAR<464,0,YPOS,0,ILINE<30>,6,0,0>
0492 C IF<LVERSN> GO TO 41
0493 C IX = 25
0494 C READ<LU,90>IANS
0495 C IF<IANS.EQ.1H0>STOP
0496 C IF<IANS .NE. 1HN> GO TO 43
0497 C IX = 30
0498 C GO TO 43
0499 C CONTINUE
0500 C CALL IN<1,JTYPE,0,0,0,0,0,0,0,31,IX,IY>
0501 C CONTINUE
0502 C CALL CHAR<0,0,YPOS,0,ILINE,64,0,0>
0503 C IF<IX .LE. 25>CALL CHAR<464,0,YPOS,0,IERS,6,0,0>
0504 C IF<IX .GT. 25>CALL CHAR<394,0,YPOS,0,IERS,6,0,0>
0505 C YPOS = YPOS - 32.0
0506 C IF<IX .LE. 25>GO TO 45
0507 C CALL DREAD<NAMEF,7,ILINE>
0508 C CALL LERS<YPOS,LVERSN>
0509 C CALL CHAR<0,0,YPOS,0,ILINE,51,3,0>
0510 C NIN = 4
0511 C CALL BLANK<IDATAF,2>
0512 C CALL IN<0,JTYPE,360,0,YPOS,0,IDATAF,NIN,0,31,IX,IY>
0513 C CALL CODE
0514 C READ<IDATAF,102> SIGAZ
0515 C CALL CODE
0516 C WRITE<ILIN24,102> SIGAZ
0517 C CALL CHAR<0,0,YPOS,0,ILINE,50,0,0>
0518 C YPOS = YPOS - 32.0
0519 C
0520 C WRITE OUT SIGAZ, THE SIGMA OF THE WIND AZIMUTH ANGLE
0521 C
0522 C 45 WRITE<LPRINT,200> SIGAZ
0523 C
0524 C SIGAP = 0.0087266 * SIGAZ
0525 C
0526 C CALCULATE THE HORIZONTAL AND VERTICAL CLOUD DIMENSIONS,
0527 C I.E. SIGX0 AND GSPEED
0528 C
0529 C SIGX0 = 0.297674 * STBALT
0530 C GSPEED = 0.232558 * STBALT
0531 C
0532 C CALCULATE AND WRITE OUT THE EFFECTIVE CLOUD HEIGHT, STBHT
0533 C
0534 C STBHT = STBALT
0535 C CLDRAD = 2.15 * SIGX0
0536 C IVZ = 0
0537 C IF<CLDRAD<STBALT .GE. ACT<LAYTOP>>IVZ = 1
0538 C SIGZ0 = SIGX0
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0539 IF<IV2 .EQ. 1>SIG20 = <ALT<LAYTOP> - STBALT + CLDRAD>/4.13  
0540 IF<SIG20 .GT. 0.0>GO TO 47  
0541 STBHT = 0.5 * ALT<LAYTOP>  
0542 SIG20 = 0.64 * STBHT/2.15  
0543 GO TO 49  
0544 IF<IV2 .EQ. 1>STBHT = 0.5 * <ALT<LAYTOP> + STBALT - CLDRAD>  
0545 C  
0546 C  
0547 C  
0548 C  
0549 C  
0550 C  
0551 C  
0552 C  
0553 C  
0554 C  
0555 C  
0556 C  
0557 C  
0558 C  
0559 C  
0560 C  
0561 C
```

47 IF<IV2 .EQ. 1>STBHT = 0.5 \* <ALT<LAYTOP> + STBALT - CLDRAD>  
49 WRITE <LUPRNT,209> STBHT  
C CALL THE CONCENTRATION PROGRAM RCONC  
C  
C CALL MGRAF  
C CALL RMDIS<NAME,1>  
C CALL EXEC<9,RCONC>  
C  
C TERMINATE EXECUTION OF RCLDS  
C  
C 1000 STOP  
C  
C END OF RCLDS  
C  
C END  
C  
C END\*

APPLIC T=0003 IS ON CR0003 USING 66118 BLKS R=6660

6661 FTMAX,L  
SUBROUTINE RUDIS(NAME, JJ), \*REDS, CAB010015

6662 C  
6663 C  
6664 C  
6665 C  
6666 C - THIS SUBROUTINE READS AND WRITES COMMON AREAS TO DISC  
6667 C - TO BE UTILIZED BY THE VARIOUS REED PROGRAMS.  
6668 C -  
6669 C -  
6670 C -

6671 C >>> COMMON AREAS

6672 C >>> MATH PARAMETERS  
COMMON PI, PIGVR2, P143, TUOP1, SQR2PI, DEGRAD, RADDEG, IV2

6673 C >>> VEHICLE PARAMETERS  
COMMON VPAK(10), SIGHCL

6674 C >>> OPTION PARAMETERS  
COMMON VPO9, YES, ISHGF0, IWHICL, IRUN, ICALC, IPLACE, ITIMEZ, IGTMP,  
NUMKUN, LU, NUM, MONLY, MFGAM, LUPRNT, ITDU, IPL1, IFL2, IFL3  
INTEGER YES

6675 C >>> MET DATA  
COMMON ALT(30), IDIR(30), SPEED(30), TEMP(30), PRESS(30), PTEMP(30),  
SURFCH, FILE(3)  
INTEGER FILE

6676 C >>> CALCULATION TIME  
COMMON ITIME, IDAY, IMONTH(2), IYEAR

6677 C >>> SOUNDING TIME  
COMMON ISTIME, ISDAY, ISMONK(2), ISYEAR

6678 C >>> LAUNCH TIME  
COMMON LTIME, LTIN, LDAY, LMONTH(2), LYEAR, LAUNTD(10)  
LOGICAL LTIME

6679 C >>> CLOUD STABILIZATION AND LAYER PARAMETERS  
COMMON STALM, STANT, STABZD, STABZC, STABIN, CLDRAD, RISTIM(30), BOTLAY,  
TOPLAY, CALHT, LAYTOP, LAYBOT, REFLEC, IPTZ

6680 C >>> DIFFUSION COEFFICIENTS  
COMMON SIGX0, SIGX, SIGZ0, SIGAP, XDIST, YDIST, EXPZ, SIGSGZ, SIGY, SIGAL,  
CLDING, AYSPD, AVDIR, SIGAZ

6681 C >>> CONCENTRATION AND DOSAGE VALUES AND RANGES  
COMMON CONMAX, CONE, PK, DOSMAX, DOSPY, RINGSMC, RINGSMD

6682 C >>> EQUIVALENCE STATEMENT FOR VEHICLE PARAMETERS  
EQUIVALENCE (GC1, VPAK(1)), (GC2, VPAK(2)), (GC3, VPAK(3)),  
(OT1, VPAK(4)), (OT2, VPAK(5)), (OT3, VPAK(6)),  
(AA, VPAK(7)), (BB, VPAK(8)), (CC, VPAK(9)),  
(HEATN, VPAK(10)), (HEATA, VPAK(11)), (HEATB, VPAK(12)),  
(HICL, VPAK(13)), (PCD, VPAK(14)), (PCD2, VPAK(15)),  
(PAI203, VPAK(16)), (CPNG, VPAK(17)), (GAMMAX, VPAK(18)),  
INTEGER DGRF(144), OBINT(557)

ORIGINAL PAGE IS  
OF POOR QUALITY

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0659 DIMENSION NAME(3) , ISIZE(2)
0660 EQUIVALENCE (OBUF(1),PI)
0661
0662 CBJ *** ISIZE(1) = 0 BLOCKS EXPECTED TO BE WRITTEN READ DURING RUN
0663 CBJ *** ISIZE(2) = 0 WORDS IN THE COMMON BLOCK TO BE TRANSFERRED
0664 DATA ISIZE/30,557/
0665
0666 CBJ *** IF THE SCRATCH DATA FILE (I.E. =REED2) ISN'T THERE, CREATE IT
0667 CBJ *** B E U A R E I I BE SURE FILE IS PURGED BEFORE STARTING RUN !!
0668 CALL OPEN(OOCB,IERR,NAME,0)
0669 IF( IERR.LT.0) .AND. (IERR.NE.-6) ) WRITE(6,770) IERR
0670 770 FORMAT( ' *** ATTEMPT TO OPEN, IERR =',14)
0671 IF( IERR.LT.0)
0672 * CALL CREAT(OOCB,IERR,NAME,ISIZE,3,0,0,144)
0673 IF( IERR.LE.0) WRITE(6,759) IERR
0674 759 FORMAT( ' *** CREATE/OPEN, IERR =',14)
0675
0676 GBJ *** READ OR WRITE, AS PER ORDER
0677 IF( J.J.EQ.1) CALL WRITF(OOCB,IERR,OBUF,ISIZE(2))
0678 IF( J.J.EQ.0) CALL READF(OOCB,IERR,OBUF,ISIZE(2))
0679 CALL CLOSE(OOCB,IERR)
0680 RETURN
0681 END
0682 SUBROUTINE FTOP
0683 C
0684 C -----
0685 C
0686 C - THIS SUBROUTINE COMPUTES THE INDEX OF THE TOP OF THE
0687 C - SURFACE LAYER BY FINDING SIGNIFICANT CHANGES IN THE
0688 C - SLOPE OF THE WIND DIRECTION, WIND SPEED, AND DRY
0689 C - TEMPERATURE VS. ALTITUDE CURVES
0690 C -
0691 C -----
0692 C
0693 C >>> COMMON AREAS
0694 C
0695 C >>> MATH PARAMETERS
0696 C COMMON PI,PIOVK2,PI43,TWOPI,SQR2PI,DEGRAD,RADDEG,IV2
0697 C
0698 C >>> VEHICLE PARAMETERS
0699 C COMMON VPAR(10),SICHL
0700 C
0701 C >>> OPTION PARAMETERS
0702 C COMMON YPOS,YES,ISMOFG,IWHICL,IRUN,ICALC,IPLACE,ITJMEZ,IGOTMP,
0703 * MUMON,LU,NUM,MOHLY,MFORM,LUPRINT,ITDU,IPL1,IPL2,IPL3
0704 C
0705 C >>> INTEGER YES
0706 C
0707 C >>> MET DATA
0708 C COMMON ALT(30),IDIR(30),SPEED(30),TEMP(30),PRESS(30),PTEMP(30),
0709 * SURFEN,FILE(3)
0710 C
0711 C >>> CALCULATION TIME
0712 C COMMON ITIME,IDAY,IMONTH(2),IYEAR
0713 C
0714 C >>> SOUNDING TIME
0715 C COMMON ISIME,ISDAY,ISMONK(2),ISYEAR
0716 C
0717 C >>> Latib... TIME
0718 C
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0119 COMMON LTIME,LTIM,LDAY,LMONTH(2),LYEAR,LAHHTD(10)
0120 LOGICAL LTIME
0121 C
0122 C >>> CLOUD STABILIZATION AND LAYER PARAMETERS
0123 COMMON STBLT,STBLT,STB4Z0,STBRG,STBTM,CLDGR0,RISTJK(30),BOTLAY,
0124 TOPLAY,CAHT,LAYTOP,LAYBOT,REFLEC,IP1Z
0125 C
0126 C >>> DIFFUSION COEFFICIENTS
0127 COMMON SIGX,SIGX,SIGZ0,SIGAF,XDIST,YDIST,EXPZ,SOSIGZ,SIGY,SIGAL,
0128 CLDNG,RYSPD,AVDIR,SIGAF
0129 C
0130 C >>> CONCENTRATION AND USAGE VALUES AND RANGES
0131 COMMON CONMAX,CONPK,DO1MAX,DO5PK,RHC3MC,RHC5MD
0132 C
0133 C >>> EQUIVALENCE STATEMENT FOR VEHICLE PARAMETERS
0134 EQUIVALENCE (OC1,VPAR(1)),(OC2,VPAR(2)),(OC3,VPAR(3)),
0135 (OT1,VPAR(4)),(OT2,VPAR(5)),(OT3,VPAR(6)),
0136 (AA,VPAR(7)),(BB,VPAR(8)),(CC,VPAR(9)),
0137 (HEATH,VPAR(10)),(HEATM,VPAR(11)),(HEATA,VPAR(12)),
0138 (PHCL,VPAR(13)),(PCO,VPAR(14)),(PCO2,VPAR(15)),
0139 (PAL203,VPAR(16)),(PHG,VPAR(17)),(GAMMAX,VPAR(18))
0140 C
0141 C DIMENSION STATEMENT
0142 C DIMENSION IPT(40),ITHET(40)
0143 C
0144 C DATA STATEMENT FOR THE SIGNIFICANT SLOPE CHANGES FOR EACH CURVE
0145 C DATA WDSLD/0.030/, US9LD/0.015/, DTSLD/0.015/
0146 C
0147 C CALCULATE THE BOTTOM OF THE CLOUD --- THE TOP OF THE
0148 C SURFACE LAYER WILL NEVER BE ALLOWED TO BE BELOW THE
0149 C BOTTOM OF THE CLOUD
0150 C
0151 C CLUBOT = STBLT * (1.0 - GAMMAX)
0152 C
0153 C
0154 C CALCULATE AND STORE THE INDEX FOR THE SIGNIFICANT SLOPE
0155 C CHANGES OF THE WIND DIRECTION VS. ALTITUDE CURVE
0156 C
0157 C DO 17 K=1,NUM
0158 ITHET(K) = IDIR(K)
0159 CONTINUE
0160 NPT = 0
0161 NUM1 = NUM - 1
0162 DELD = FLOAT(ITHET(2) - ITHET(1))
0163 SLOPE2 = DELD/(ALT(2) - ALT(1))
0164 DO 3 I=2,NUM1
0165 SLOPE1 = SLOPE2
0166 DELD = FLOAT(ITHET(I+1) - ITHET(I))
0167 SLOPE2 = DELD/(ALT(I+1) - ALT(I))
0168 IF(ALT(1) .LT. CLUBOT)GO TO 3
0169 SLDIF = ABS(SLOPE2 - SLOPE1)
0170 IF(SLDIF .LT. WDSLD)GO TO 3
0171 NPT = NPT + 1
0172 IPT(NPT) = I
0173 CONTINUE
0174 C
0175 C CALCULATE AND STORE THE INDEX FOR THE SIGNIFICANT SLOPE
0176 C CHANGES OF THE WIND SPEED VS. ALTITUDE CURVE
0177 C
0178 C
```

ORIGINAL PAGE IS  
OF POOR QUALITY.

```
0175 SLOPE2 = (SPEED(2) - SPEED(1))/(ALT(2) - ALT(1))
0180 DO 6 I=2,NUM1
0181 SLOPE1 = SLOPE2
0182 SLOPE2 = (SPEED(I+1) - SPEED(I))/(ALT(I+1) - ALT(I))
0183 IF(ALT(I) .LT. CLOBOT)GO TO 6
0184 SLDIF = ABS(SLOPE2 - SLOPE1)
0185 IF(SLDIF .LT. WSSLD)GO TO 6
0186 NPT = NPT + 1
0187 IPT(NPT) = I
0188 6 CONTINUE
0189 C
0190 C CALCULATE AND STORE THE INDEX FOR THE SIGNIFICANT SLOPE
0191 C CHANGES OF THE DRY TEMPERATURE VS. ALTITUDE CURVE
0192 C
0193 SLOPE2 = (TEMP(2) - TEMP(1))/(ALT(2) - ALT(1))
0194 DO 9 I=2,NUM1
0195 SLOPE1 = SLOPE2
0196 SLOPE2 = (TEMP(I+1) - TEMP(I))/(ALT(I+1) - ALT(I))
0197 IF(ALT(I) .LT. CLOBOT)GO TO 9
0198 SLDIF = ABS(SLOPE2 - SLOPE1)
0199 IF(SLDIF .LT. DTSLD)GO TO 9
0200 NPT = NPT + 1
0201 IPT(NPT) = I
0202 9 CONTINUE
0203 C
0204 C IF THERE ARE NO SIGNIFICANT SLOPE CHANGES, MAKE THE
0205 C TOP OF THE LAYER TWICE THE CLOUD STABILIZATION ALTITUDE
0206 C
0207 IF(NPT .GT. 0)GO TO 14
0208 STBAL2 = 2.0 * STBAL1
0209 DIFMIN = 9999999.9
0210 DO 12 I=1,NUM
0211 DIF = ABS(ALT(I) - STBAL2)
0212 IF(DIF .GE. DIFMIN)GO TO 12
0213 DIFMIN = DIF
0214 LAYTOP = I
0215 12 CONTINUE
0216 RETURN
0217 C
0218 C IF THERE IS ONLY ONE SIGNIFICANT SLOPE CHANGE, MAKE IT THE
0219 C TOP OF THE SURFACE LAYER
0220 C
0221 IF(NPT .GT. 1)GO TO 16
0222 LAYTOP = IPT(1)
0223 RETURN
0224 C
0225 C SORT THE SIGNIFICANT SLOPE CHANGE POINTS FROM HIGH TO LOW
0226 C
0227 NPT1 = NPT - 1
0228 DO 19 I=1,NPT1
0229 JJ = NPT - I
0230 DO 18 J=1, JJ
0231 J1 = J + 1
0232 IF(IPT(J) .GE. IPT(J1))GO TO 18
0233 K = IPT(J)
0234 IPT(J) = IPT(J1)
0235 IPT(J1) = K
0236 18 CONTINUE
0237 19 CONTINUE
0238
```

ORIGINAL PAGE IS  
OF POOR QUALITY

0239 C FIRST CHECK FOR SIGNIFICANT SLOPE CHANGES FROM ALL THREE  
0240 C CURVES AT THE SAME POINT -- NEXT CHECK FOR SIGNIFICANT  
0241 C SLOPE CHANGES FROM TWO OUT OF THE THREE CURVES AT THE SAME  
0242 C POINT -- THE FIRST POINT MEETING ONE OF THESE REQUIREMENTS  
0243 C WILL BE THE TOP OF THE SURFACE LAYER  
0244 C

0245 DO 27 I=1,2  
0246 JJ = 4 - I  
0247 JI = 1  
0248 DO 25 J=2,MP7  
0249 IF(IPT<J) .EQ. IPT<J-1)GO TO 22  
0250 JI = 1

0251 GO TO 25  
0252 22 JI = JI + 1  
0253 IF<JI .NE. JJ)GO TO 25  
0254 LAYTOP = IPT<J)  
0255 RETURN

0256 25 CONTINUE  
0257 27 CONTINUE

0258 C THERE ARE NO DUPLICATE POINTS OF SIGNIFICANT SLOPE CHANGE,  
0259 C MAKE THE TOP OF THE SURFACE LAYER THE FIRST POINT  
0260 C

0261 C LAYTOP = IPT<1)  
0262 C RETURN

0263 C  
0264 C  
0265 C END OF FTOP  
0266 C

0267 C  
0268 C END  
0269 C SUBROUTINE PTTOP

0270 C  
0271 C  
0272 C  
0273 C  
0274 C  
0275 C  
0276 C  
0277 C

0278 C --- THIS SUBROUTINE ALLOWS THE USER TO CHOOSE THE TOP OF THE ---  
0279 C SURFACE LAYER BY WRITING SIGNIFICANT CHANGES IN THE ---  
0280 C SLOPE OF THE WIND DIRECTION, WIND SPEED, AND DRY ---  
0281 C TEMPERATURE VS. ALTITUDE CURVES TO THE CRT OR PLASMA3SCOPE ---  
0282 C  
0283 C  
0284 C

0285 C >>> COMMON AREAS  
0286 C

0287 C >>> MATH PARAMETERS  
0288 C

0289 C COMMON PI,P10VR2,F143,TU0PI,SAR2PI,DEGRAD,RADDEG,IV2  
0290 C

0291 C >>> VEHICLE PARAMETERS  
0292 C COMMON VPAK<18>,SIGH<1

0293 C >>> OPTION PARAMETERS  
0294 C

0295 C COMMON YPOS,YES,ISINFO,IVHICL,IRUN,ICALC,IPLACE,ITIME2,IGOTMP,  
0296 C NUMRUN,LU,NUM,MOHLY,MFORN,LUFANT,ITDU,IPL1,IPL2,IPL3  
0297 C INTEGER YES

0298 C >>> MET DATA  
0299 C

0300 C COMMON ALT<30>,IDIR<30>,SPEED<30>,TEMP<30>,PRESS<30>,PTEMP<30>,  
0301 C SUPDEN,FILE<3>  
0302 C INTEGER FILE

0303 C >>> CALCULATION TIME  
0304 C

ORIGINAL PAGE IS  
OF POOR QUALITY

```

COMMON I TIME, IDAY, IMONTH(2), IYEAR
C
0299 C
0300 C
0301 C >>> SOUNDING TIME
0302 C COMMON I TIME, ISDAY, ISMON(2), ISYEAR
0303 C
0304 C >>> LAUNCH TIME
0305 C COMMON L TIME, LTIM, LDAY, LMONTH(2), LYEAR, LAUNTD(10)
0306 C LOGICAL L TIME
C
0307 C
0308 C >>> CLOUD STABILIZATION AND LAYER PARAMETERS
0309 C COMMON STBAL, STBHT, STBAZD, STBRNG, STBTIM, CLORAD, RISTIM(30), BOTLTY,
0310 C TOPLAY, CALHT, LAYTOP, LAYBOT, REFLEC, IPTZ
0311 C
0312 C >>> DIFFUSION COEFFICIENTS
0313 C COMMON SIGX, SIGY, SIGZ0, SIGAF, XDIST, YDIST, EXPZ, SOSIGZ, SIGY, SIGAL,
0314 C CLDINC, WSPD, AVDIR, SIGAZ
C
0315 C
0316 C >>> CONCENTRATION AND DOSAGE VALUES AND RANGES
0317 C COMMON CONMAX, CONCPK, DOSMAX, DOSPK, RINGSMC, RINGSMD
0318 C
0319 C >>> EQUIVALENCE STATEMENT FOR VEHICLE PARAMETERS
0320 C EQUIVALENCE (QC1, VPAR(1)), (QC2, VPAR(2)), (QC3, VPAR(3)),
0321 C (QT1, VPAR(4)), (QT2, VPAR(5)), (QT3, VPAR(6)),
0322 C (AA, VPAR(7)), (AB, VPAR(8)), (CC, VPAR(9)),
0323 C (HEATN, VPAR(10)), (HEATH, VPAR(11)), (HEATA, VPAR(12)),
0324 C (PHCL, VPAR(13)), (PCO, VPAR(14)), (PCO2, VPAR(15)),
0325 C (PAL203, VPAR(16)), (PHO, VPAR(17)), (GAMMAX, VPAR(18))
0326 C
0327 C DIMENSION STATEMENT
0328 C
0329 C DIMENSION IPT(40), DMDS(40), WSDS(40), TDS(40), ITHET(40),
0330 C DSTAR(39), SSTAR(39), TSTAR(39)
C
0331 C
0332 C DATA STATEMENT FOR THE SIGNIFICANT SLOPE CHANGES FOR EACH CURVE
0333 C DATA WSSLD/0.030/, WSSLD/0.015/, DTSLD/0.015/,
0334 C DATA ISTR/INH*/
0335 C
0336 C
0337 C CALCULATE THE BOTTOM OF THE CLOUD -- THE TOP OF THE
0338 C SURFACE LAYER WILL NEVER BE ALLOWED TO BE BELOW THE
0339 C BOTTOM OF THE CLOUD
0340 C
0341 C CLDBOT = STBAL * (1.0 - GAMMAX)
0342 C STBAL2 = 2.0 * STBAL
0343 C
0344 C ZERO OUT THE SIGNIFICANT SLOPE ARRAYS
0345 C
0346 C DO 5 K=1, NUM
0347 C DMDS(K) = 0.0
0348 C WSDS(K) = 0.0
0349 C TDS(K) = 0.0
0350 C DSTAR(K) = 0.0
0351 C SSTAR(K) = 0.0
0352 C TSTAR(K) = 0.0
C
0353 C
0354 C CALCULATE AND STORE THE INDEX FOR THE SIGNIFICANT SLOPE
0355 C CHANGES OF THE WIND DIRECTION VS. ALTITUDE CURVE
0356 C
0357 C
0358 C

```



```

0359 17 CONTINUE
0360 HPT = 0
0361 NUM1 = NUM - 1
0362 DELD = FLOAT(ITHET<2>) - ITHET<1>
0363 IF<DELD.GT.100.> DELD = 360. - DELD
0364 IF<DELD.LT.-100.> DELD = -360. - DELD
0365 SLOPE2 = DELD/<ALT<2> - ALT<1>
0366 C
0367 DO 3 I=2,NUM1
0368 SLOPE1 = SLOPE2
0369 DELD = FLOAT(ITHET<I+1>) - ITHET<I>
0370 IF<DELD.GT.100.> DELD = 360. - DELD
0371 IF<DELD.LT.-100.> DELD = -360. - DELD
0372 SLOPE2 = DELD/<ALT<I+1> - ALT<I>
0373 IF<ALT<I> .LT. CLOUDOT .OR. ALT<I> .GT. STBAL2> GO TO 3
0374 SLDIF = <SLOPE2 - SLOPE1>
0375 IF<ABS<SLDIF> .LT. 0.01> GO TO 3
0376 IF<SLOPE2.GT.0.0.AND.SLOPE1.LT.0.0> DSTAR<I> = SLDIF
0377 IF<SLOPE1.GT.0.0.AND.SLOPE2.LT.0.0> DSTAR<I> = SLDIF
0378 HPT = HPT + 1
0379 IPT<HPT> = 1
0380 DSDS<I> = SLDIF
0381 C
0382 3 CONTINUE
0383 C
0384 C CALCULATE AND STORE THE INDEX FOR THE SIGNIFICANT SLOPE
0385 C CHANGES OF THE WIND SPEED VS. ALTITUDE CURVE
0386 C
0387 SLOPE2 = <SPEED<2> - SPEED<1>>/<ALT<2> - ALT<1>
0388 DO 6 I=2,NUM1
0389 SLOPE1 = SLOPE2
0390 SLOPE2 = <SPEED<I+1> - SPEED<I>>/<ALT<I+1> - ALT<I>
0391 IF<ALT<I> .LT. CLOUDOT .OR. ALT<I> .GT. STBAL2> GO TO 6
0392 SLDIF = <SLOPE2 - SLOPE1>
0393 IF<ABS<SLDIF> .LT. 0.001> GO TO 6
0394 IF<SLOPE1.GT.0.0.AND.SLOPE2.LT.0.0> SSTAR<I> = SLDIF
0395 IF<SLOPE2.GT.0.0.AND.SLOPE1.LT.0.0> SSTAR<I> = SLDIF
0396 WSDS<I> = SLDIF
0397 HPT = HPT + 1
0398 IPT<HPT> = 1
0399 C
0400 6 CONTINUE
0401 C
0402 C CALCULATE AND STORE THE INDEX FOR THE SIGNIFICANT SLOPE
0403 C CHANGES OF THE POTENTIAL TEMPERATURE VS. ALTITUDE CURVE
0404 C
0405 SLOPE2 = <TEMP<2> - TEMP<1>>/<ALT<2> - ALT<1>
0406 DO 9 I=2,NUM1
0407 SLOPE1 = SLOPE2
0408 SLOPE2 = <TEMP<I+1> - TEMP<I>>/<ALT<I+1> - ALT<I>
0409 IF<ALT<I> .LT. CLOUDOT .OR. ALT<I> .GT. STBAL2> GO TO 9
0410 SLDIF = <SLOPE2 - SLOPE1>
0411 IF<ABS<SLDIF> .LT. 0.01> GO TO 9
0412 IF<SLOPE1.GT.0.0.AND.SLOPE2.LT.0.0> TSTAR<I> = SLDIF
0413 IF<SLOPE2.GT.0.0.AND.SLOPE1.LT.0.0> TSTAR<I> = SLDIF
0414 HPT = HPT + 1
0415 IPT<HPT> = 1
0416 TDS<I> = SLDIF
0417 C
0418 9 CONTINUE
0419 C
0420 C IF THERE ARE NO SIGNIFICANT SLOPE CHANGES, KEEP THE
0421 C TOP OF THE LAYER TWICE THE CLOUD STABILIZATION ALTITUDE

```



```

0479 WRITE (LU,103) K,ALT(K),DWDS(K),USDS(K),TDS(K)
0480 IF(DSTAR(K).NE.0.0.AND.SSTAR(K).NE.0.0.AND.TSTAR(K).EQ.0.0)
0481 WRITE (LU,104) K,ALT(K),DWDS(K),ISTR,USDS(K),ISTR,TDS(K)
0482 IF(DSTAR(K).NE.0.0.AND.SSTAR(K).NE.0.0.AND.TSTAR(K).NE.0.0)
0483 WRITE (LU,105) K,ALT(K),DWDS(K),ISTR,USDS(K),ISTR,TDS(K),ISTR
0484 IF(DSTAR(K).NE.0.0.AND.SSTAR(K).EQ.0.0.AND.TSTAR(K).EQ.0.0)
0485 WRITE (LU,106) K,ALT(K),DWDS(K),ISTR,USDS(K),TDS(K)
0486 IF(DSTAR(K).EQ.0.0.AND.SSTAR(K).NE.0.0.AND.TSTAR(K).EQ.0.0)
0487 WRITE (LU,107) K,ALT(K),DWDS(K),USDS(K),ISTR,TDS(K)
0488 IF(DSTAR(K).EQ.0.0.AND.SSTAR(K).EQ.0.0.AND.TSTAR(K).NE.0.0)
0489 WRITE (LU,108) K,ALT(K),DWDS(K),USDS(K),TDS(K),ISTR
0490 IF(DSTAR(K).EQ.0.0.AND.SSTAR(K).NE.0.0.AND.TSTAR(K).NE.0.0)
0491 WRITE (LU,109) K,ALT(K),DWDS(K),USDS(K),ISTR,TDS(K),ISTR
0492 IF(DSTAR(K).NE.0.0.AND.SSTAR(K).EQ.0.0.AND.TSTAR(K).NE.0.0)
0493 WRITE (LU,110) K,ALT(K),DWDS(K),ISTR,USDS(K),TDS(K),ISTR
0494
0495 C** WRITE THE CRT DISPLAY TO THE PRINTER.
0496 C
0497 IF(DSTAR(K).EQ.0.0.AND.SSTAR(K).EQ.0.0.AND.TSTAR(K).EQ.0.0)
0498 WRITE (6,103) K,ALT(K),DWDS(K),USDS(K),TDS(K)
0499 IF(DSTAR(K).NE.0.0.AND.SSTAR(K).NE.0.0.AND.TSTAR(K).EQ.0.0)
0500 WRITE (6,104) K,ALT(K),DWDS(K),ISTR,USDS(K),ISTR,TDS(K)
0501 IF(DSTAR(K).NE.0.0.AND.SSTAR(K).NE.0.0.AND.TSTAR(K).NE.0.0)
0502 WRITE (6,105) K,ALT(K),DWDS(K),ISTR,USDS(K),ISTR,TDS(K),ISTR
0503 IF(DSTAR(K).NE.0.0.AND.SSTAR(K).EQ.0.0.AND.TSTAR(K).EQ.0.0)
0504 WRITE (6,106) K,ALT(K),DWDS(K),ISTR,USDS(K),TDS(K)
0505 IF(DSTAR(K).EQ.0.0.AND.SSTAR(K).NE.0.0.AND.TSTAR(K).EQ.0.0)
0506 WRITE (6,107) K,ALT(K),DWDS(K),USDS(K),ISTR,TDS(K)
0507 IF(DSTAR(K).EQ.0.0.AND.SSTAR(K).EQ.0.0.AND.TSTAR(K).NE.0.0)
0508 WRITE (6,108) K,ALT(K),DWDS(K),USDS(K),TDS(K),ISTR
0509 IF(DSTAR(K).EQ.0.0.AND.SSTAR(K).NE.0.0.AND.TSTAR(K).NE.0.0)
0510 WRITE (6,109) K,ALT(K),DWDS(K),USDS(K),ISTR,TDS(K),ISTR
0511 IF(DSTAR(K).NE.0.0.AND.SSTAR(K).EQ.0.0.AND.TSTAR(K).NE.0.0)
0512 WRITE (6,110) K,ALT(K),DWDS(K),ISTR,USDS(K),TDS(K),ISTR
0513 33 CONTINUE
0514 FORMAT(16,F8.1,F12.5,A1,F11.5,A1,F12.5)
0515 FORMAT(16,F8.1,F12.5,A1,F11.5,A1,F11.5,A1)
0516 FORMAT(16,F8.1,F12.5,A1,F12.5,A1,F13.5)
0517 FORMAT(16,F8.1,F12.5,F13.5,A1,F12.5)
0518 FORMAT(16,F8.1,F12.5,F13.5,A1)
0519 FORMAT(16,F8.1,F12.5,F13.5,A1,F12.5,A1)
0520 110 FORMAT(16,F8.1,F12.5,A1,F12.5,F13.5,A1)
0521
0522 WRITE (LU,112)
0523 WRITE (6,112)
0524 112 FORMAT(" * INDICATES A REVERSAL IN THE SIGN OF THE SLOPE")
0525 34 WRITE (LU,111)
0526 111 FORMAT(" &dJ ENTER THE NUMBER OF THE LAYER FOR THE TOP &de-
" &dJ OF THE SURFACE LAYER&do")
0527 READ (LU,*) LAYER
0528 LAYTOP = LAYER
0529 RETURN
0530
0531 C
0532 C
0533 C
0534 C
0535 C
0536 C
0537 C
0538 C
0539 C
0540 C
0541 C
0542 C
0543 C
0544 C
0545 C
0546 C
0547 C
0548 C
0549 C
0550 C
0551 C
0552 C
0553 C
0554 C
0555 C
0556 C
0557 C
0558 C
0559 C
0560 C
0561 C
0562 C
0563 C
0564 C
0565 C
0566 C
0567 C
0568 C
0569 C
0570 C
0571 C
0572 C
0573 C
0574 C
0575 C
0576 C
0577 C
0578 C
0579 C
0580 C
0581 C
0582 C
0583 C
0584 C
0585 C
0586 C
0587 C
0588 C
0589 C
0590 C
0591 C
0592 C
0593 C
0594 C
0595 C
0596 C
0597 C
0598 C
0599 C
0600 C
0601 C
0602 C
0603 C
0604 C
0605 C
0606 C
0607 C
0608 C
0609 C
0610 C
0611 C
0612 C
0613 C
0614 C
0615 C
0616 C
0617 C
0618 C
0619 C
0620 C
0621 C
0622 C
0623 C
0624 C
0625 C
0626 C
0627 C
0628 C
0629 C
0630 C
0631 C
0632 C
0633 C
0634 C
0635 C
0636 C
0637 C
0638 C
0639 C
0640 C
0641 C
0642 C
0643 C
0644 C
0645 C
0646 C
0647 C
0648 C
0649 C
0650 C
0651 C
0652 C
0653 C
0654 C
0655 C
0656 C
0657 C
0658 C
0659 C
0660 C
0661 C
0662 C
0663 C
0664 C
0665 C
0666 C
0667 C
0668 C
0669 C
0670 C
0671 C
0672 C
0673 C
0674 C
0675 C
0676 C
0677 C
0678 C
0679 C
0680 C
0681 C
0682 C
0683 C
0684 C
0685 C
0686 C
0687 C
0688 C
0689 C
0690 C
0691 C
0692 C
0693 C
0694 C
0695 C
0696 C
0697 C
0698 C
0699 C
0700 C

```

```

0539 C - THIS SUBROUTINE READS A SPECIFIED QUESTION FROM A SPECIFIED
0540 C - QUESTION FILE. THE QUESTION IS THEN DISPLAYED UPON THE
0541 C - PLASMASCOPE FOR PROCESSING.
0542 C -----
0543 C -----
0544 C
0545 C DIMENSION NAMEF(3),IDCB(144),IBUF(40),ILINE(32)
0546 C CALL OPEN(IDCB,IERR,NAMEF,0)
0547 C IF(IERR.NE.0)WRITE(6,799) IERR, NAMEF
0548 799 FORMAT(' ERROR ',I5, ' ON FILE ',3A2)
0549 C LOOP = LNUM - 1
0550 C DO 10 I=1,LOOP
0551 C CALL BLANK(IBUF,40)
0552 C CALL READF(IDCB,IERR,IBUF)
0553 C 10 CONTINUE
0554 C CALL BLANK(IBUF,40)
0555 C CALL READF(IDCB,IERR,IBUF)
0556 C CALL CODE
0557 C READ (IBUF,100) (ILINE(I),I=1,32)
0558 100 FORMAT(32A2)
0559 C CALL CLOSE(IDCB,IERR)
0560 C RETURN
0561 C END
0562 C SUBROUTINE BLANK(IBUF,II)
0563 C -----
0564 C -----
0565 C - THIS SUBROUTINE FILLS A SPECIFIED ARRAY WITH A SPECIFIED
0566 C - NUMBER OF BLANKS.
0567 C -----
0568 C -----
0569 C -----
0570 C
0571 C DIMENSION IBUF(40)
0572 C DATA IBLK/2H /
0573 C DO 10 I=1,II
0574 10 IBUF(I) = IBLK
0575 C RETURN
0576 C END
0577 C SUBROUTINE LERS(YPOS)
0578 C -----
0579 C -----
0580 C - THIS SUBROUTINE ERASES A SPECIFIED LINE OF 64-CHARACTERS
0581 C - FROM THE PLASMASCOPE.
0582 C -----
0583 C -----
0584 C -----
0585 C
0586 C DIMENSION IERS(32)
0587 C DATA IERS/32+2H /
0588 C IF(YPOS.GT.48.0)GO TO 4
0589 C YPOS = 458.0
0590 C CALL CLEAR
0591 C RETURN
0592 C 4 CALL CHAR(0.0,YPOS,0,IERS,64,0,0)
0593 C CALL CHAR(0.0,YPOS-16.0,0,IERS,64,0,0)
0594 C RETURN
0595 C END
0596 C SUBROUTINE RSGAZ(J1,J2,J3,RSIG)
0597 C -----
0598 C -----
0599 C -----
0600 C -----

```

```

0599 C - THIS SUBROUTINE CALCULATES A SIGMA VALUE GIVEN
0600 C - ALTITUDE, SPEED, TEMP, AND PRESSURE FOR THE
0601 C - FIRST LEVEL OF DATA, THE 1000FT LEVEL OF DATA
0602 C - AND THE 1000MB LEVEL OF DATA
0603 C
0604 C -----
0605 C
0606 C >>> COMMON AREAS
0607 C
0608 C
0609 C >>> MATH PARAMETERS
0610 C COMMON PI,PIOV2,P143,TUOP1,SOR2P1,DEGRAD,RADEG,IV2
0611 C
0612 C >>> VEHICLE PARAMETERS
0613 C COMMON VPAK(10),SICHCL
0614 C
0615 C >>> OPTION PARAMETERS
0616 C COMMON OS,YES,ISNOFO,IWHICL,IRUN,ICALC,IPLACE,ITIMEZ,IGOTMP,
0617 C          IN,LU,NUM,MONLY,MFORM,LUPRNT,ITDU,IPL1,IPL2,IPL3
0618 C
0619 C INTEGER
0620 C
0621 C >>> MET DATA
0622 C COMMON ALI(30),IDIR(30),SPEED(30),TEMP(30),PRESS(30),PIEMP(30),
0623 C          BURDEH,FILE(3)
0624 C
0625 C >>> CALCULATION TIME
0626 C COMMON ITIME,IDAY,IMONTH(2),IYEAR
0627 C
0628 C >>> SOUNDING TIME
0629 C COMMON ISTIME,ISDAY,ISMOIK(2),ISYEAR
0630 C
0631 C >>> LAUNCH TIME
0632 C COMMON LTIME,LTIM,LDAY,LMONTH(2),LYEAR,LAUNTD(10)
0633 C
0634 C LOGICAL LTIME
0635 C
0636 C >>> CLOUD STABILIZATION AND LAYER PARAMETERS
0637 C COMMON STBLT,STBT,STBZD,STBRG,STBTM,CLDRAD,RISTIK(30),BOTLAY,
0638 C          TOPLAY,CALNT,LAYTOP,LAYBOT,REFLEC,IPTZ
0639 C
0640 C >>> DIFFUSION COEFFICIENTS
0641 C COMMON SIGX,SIGY,SIGZ0,SIGAP,XDIST,YDIST,EXPZ,SOSIGZ,SIGY,SIGAL,
0642 C          CLDLG,AVSPD,AVDIR,SIGAZ
0643 C
0644 C >>> CONCENTRATION AND DOSAGE VALUES AND RANGES
0645 C COMMON CONMAX,COMCPK,DOSMAX,DOSPK,RNGSMC,RNGSHD
0646 C
0647 C >>> EQUIVALENCE STATEMENT FOR VEHICLE PARAMETERS
0648 C EQUIVALENCE (QC1,VPAK(1)),(QC2,VPAK(2)),(QC3,VPAK(3)),
0649 C          (QT1,VPAK(4)),(QT2,VPAK(5)),(QT3,VPAK(6)),
0650 C          (AA,VPAK(7)),(BB,VPAK(8)),(CC,VPAK(9)),
0651 C          (HEATN,VPAK(10)),(HEATH,VPAK(11)),(CHEATA,VPAK(12)),
0652 C          (PHCL,VPAK(13)),(PCO,VPAK(14)),(PCO2,VPAK(15)),
0653 C          (PAL203,VPAK(16)),(PHO,VPAK(17)),(GAMMAX,VPAK(18))
0654 C
0655 C DATA C1,C2,C3,C4,C5,C6/-.008,-.00175,.0008,.56864522,.1132,
0656 C          3.8163/
0657 C
0658 C DATA C1,C2,C3,C4,C5,C6/-.008,-.00175,.0008,.56864522,.1132,
0659 C          3.8163/
0660 C
0661 C
0662 C
0663 C
0664 C
0665 C
0666 C
0667 C
0668 C
0669 C
0670 C
0671 C
0672 C
0673 C
0674 C
0675 C
0676 C
0677 C
0678 C
0679 C
0680 C
0681 C
0682 C
0683 C
0684 C
0685 C
0686 C
0687 C
0688 C
0689 C
0690 C
0691 C
0692 C
0693 C
0694 C
0695 C
0696 C
0697 C
0698 C
0699 C
0700 C
0701 C
0702 C
0703 C
0704 C
0705 C
0706 C
0707 C
0708 C
0709 C
0710 C
0711 C
0712 C
0713 C
0714 C
0715 C
0716 C
0717 C
0718 C
0719 C
0720 C
0721 C
0722 C
0723 C
0724 C
0725 C
0726 C
0727 C
0728 C
0729 C
0730 C
0731 C
0732 C
0733 C
0734 C
0735 C
0736 C
0737 C
0738 C
0739 C
0740 C
0741 C
0742 C
0743 C
0744 C
0745 C
0746 C
0747 C
0748 C
0749 C
0750 C
0751 C
0752 C
0753 C
0754 C
0755 C
0756 C
0757 C
0758 C
0759 C
0760 C
0761 C
0762 C
0763 C
0764 C
0765 C
0766 C
0767 C
0768 C
0769 C
0770 C
0771 C
0772 C
0773 C
0774 C
0775 C
0776 C
0777 C
0778 C
0779 C
0780 C
0781 C
0782 C
0783 C
0784 C
0785 C
0786 C
0787 C
0788 C
0789 C
0790 C
0791 C
0792 C
0793 C
0794 C
0795 C
0796 C
0797 C
0798 C
0799 C
0800 C
0801 C
0802 C
0803 C
0804 C
0805 C
0806 C
0807 C
0808 C
0809 C
0810 C
0811 C
0812 C
0813 C
0814 C
0815 C
0816 C
0817 C
0818 C
0819 C
0820 C
0821 C
0822 C
0823 C
0824 C
0825 C
0826 C
0827 C
0828 C
0829 C
0830 C
0831 C
0832 C
0833 C
0834 C
0835 C
0836 C
0837 C
0838 C
0839 C
0840 C
0841 C
0842 C
0843 C
0844 C
0845 C
0846 C
0847 C
0848 C
0849 C
0850 C
0851 C
0852 C
0853 C
0854 C
0855 C
0856 C
0857 C
0858 C
0859 C
0860 C
0861 C
0862 C
0863 C
0864 C
0865 C
0866 C
0867 C
0868 C
0869 C
0870 C
0871 C
0872 C
0873 C
0874 C
0875 C
0876 C
0877 C
0878 C
0879 C
0880 C
0881 C
0882 C
0883 C
0884 C
0885 C
0886 C
0887 C
0888 C
0889 C
0890 C
0891 C
0892 C
0893 C
0894 C
0895 C
0896 C
0897 C
0898 C
0899 C
0900 C

```

ORIGINAL PAGE IS  
OF POOR QUALITY

```

0659 F(X,D) = (1.-X**4)/(16.+X**2*(ALOG(D)+C4-2.*ALOG(1.+X)
0660 - ALOG(1.+X**2)+2.*ATAN(X))**2 - B
0661 FP(X,D) = (-X**4-1.)/(8.+X**3*(ALOG(D)+C4-2.*ALOG(1.+X)
0662 - ALOG(1.+X**2)+2.*ATAN(X))**2 + (1.-X**4)/(2.*(1.+X)
0663 + (1.+X**2)*(ALOG(D)+C4-2.*ALOG(1.+X)-ALOG(1.+X**2)+
0664 + 2.*ATAN(X))**3)
0665 C
0666 C*** READ 1ST DATA LEVEL
0667 C
0668 Z1 = ALT(J1)
0669 V1 = SPEED(J1)
0670 T1 = TEMP(J1)
0671 PZ1 = PRESS(J1)
0672 C
0673 C*** READ SECOND DATA LEVEL
0674 C
0675 Z2 = ALT(J2)
0676 V2 = SPEED(J2)
0677 V2 = TEMP(J2)
0678 PZ2 = PRESS(J2)
0679 C
0680 C*** IF TOWER DATA IS NOT BEING USED, INTERPOLATE TO FIND THE 1000
0681 C*** FOOT LEVEL DATA
0682 C
0683 Z3 = ALT(J3)
0684 T3 = TEMP(J3)
0685 PZ3 = PRESS(J3)
0686 WRITE (6,613) PZ1,PZ2,PZ3
0687 613 FORMAT(' PRESSURES ARE:',3F10.2)
0688 IF(IIDU.EQ.1) GO TO 5
0689 C
0690 FAC = (304.8 - ALT(J3))/(ALT(J3+1)-ALT(J3))
0691 Z3 = ALT(J3)+(FAC*(ALT(J3+1)-ALT(J3)))
0692 T3 = TEMP(J3) + FAC*(TEMP(J3+1)-TEMP(J3))
0693 PZ3 = PRESS(J3) + FAC*(PRESS(J3+1)-PRESS(J3))
0694 C
0695 S T1 = T1+273.16
0696 T2 = T2+273.16
0697 T3 = T3+273.16
0698 C
0699 C*** INITIALIZE Z0
0700 C
0701 Z0 = .20
0702 C PZ1 AND PZ3 IN MILLIBARS
0703 C V1,V2 AND V3 IN METER/SEC
0704 C Z1,Z2 AND Z3 IN METERS
0705 C T1,T2 AND T3 IN DEG K
0706 C Z0 IN METERS
0707 E = 22.9183118
0708 V=V2
0709 T=(T1+T2+T3)/3.
0710 C
0711 C*** E IS A CONSTANT INCORPORATING VON KARMAN'S CONSTANT AND
0712 C*** ALSO CONVERTS FROM RADIAN TO DEGREES.
0713 C
0714 C*** CALCULATE GEOMETRIC MEAN HEIGHT, AVERAGE THETA, AVERAGE HEIGHT
0715 C
0716 Z=(Z1+Z2+Z3)**.33333
0717 SURTAL = 1+(1000./P**).25
0718 METH2 = 12

```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0719 THETA3 = I3*((1000./P23)**.286)
0720 ZA = (Z1+Z2+Z3)/3.
0721 THETA0 = (THETA1 + THETA2 + THETA3)/3.
0722 D = Z/20
0723 Z020 = ALOG(D)
0724 C
0725 C*** CALCULATE VERTICAL GRADIENT OF POTENTIAL TEMPERATURE
0726 C
0727 DZTHET = ((Z1-ZA)**(THETA1-THETA0)+(Z2-ZA)**(THETA2-THETA0)
0728 + (Z3-ZA)**(THETA3-THETA0))/(Z1-ZA)**2 + (Z2-ZA)**2
0729 + (Z3-ZA)**2)
0730 C
0731 C*** CALCULATE B, THE STABILITY RATIO
0732 C*** B.LT.0 = UNSTABLE
0733 C*** B.GT.0 = STABLE
0734 C
0735 B = 9.0+DZTHET*Z**2/(T+V**2)
0736 IF(B.LT.0) GO TO 2
0737 IF(B.GT.0) GO TO 6
0738 RSIG = 48.016/ALOG(D)
0739 GO TO 1000
0740 C
0741 C*** USE NEWTON'S METHOD TO SOLVE EQUATION 14 (REPORT 2923)
0742 C
0743 2 R = 1.5
0744 U = F(R,B,D)
0745 DO 3 I = 1,50
0746 R1 = R - F(R,B,D)/FP(R,D)
0747 U=FC(R1,B,D)
0748 IF(ABS(R1-R).LT.1.E-7) GO TO 21
0749 IF(1.EQ.49) USAV = U
0750 IF(1.NE.50) GO TO 3
0751 IF(USAV.LT.0..AND.U.GT.0..OR.USAV.GT.0..AND.U.LT.0.) GO TO 21
0752 3 R = R1
0753 RSIG = 30.
0754 GO TO 1000
0755 21 R11 = (1.-R1)**4/16.
0756 A = Z020 +C4-2.*ALOG(1.+R1)-ALOG(1.+R1**2)+2.*ATN(R1)
0757 C
0758 C*** ASSIGN THE CORRECT F(B) FOR EACH VALUE OF B ACCORDING TO
0759 C*** TABLE 6 IN REPORT 2923
0760 C
0761 IF(B.LT.C1) RSIG = E+2.7/A
0762 IF(B.GE.C1.AND.B.LT.C2) GO TO 23
0763 IF(B.GE.C2) GO TO 24
0764 GO TO 1000
0765 C
0766 C*** COMPUTE SIGR ACCORDING TO EQUATION 1 IN REPORT 2923 FOR
0767 C*** UNSTABLE CONDITIONS
0768 C
0769 23 FB2 = 2.7 + 112.*(C1 + B)
0770 RSIG = E+FB2/A
0771 GO TO 1000
0772 24 FB3 = 3.4 - 725.5*(C2 +B)
0773 RSIG = E+FB3/A
0774 GO TO 1000
0775 C
0776 C*** FOR STABLE CONDITIONS, USE EQUATION 13(A) IN REPORT 2923 TO
0777 C*** CALCULATE RICHARDSON NUMBER
0778 C
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```

0779 6      AP = Z020 + I.
0780      A1 = 7.*SQRT(B)*AP
0781      A2 = I.
0782      A3 = -80RT(B)*(AP-I.)
0783      RAD = A2**2 - 4.*A1*A3
0784      C*** CHECK SIGN OF THE RADICAL, USE VALUE OF RSGIC = 30. FOR
0785      C*** IMAGINARY VALUES.
0786      C***
0787      C
0788      RSGIC = 30.0
0789      IF(RAD.LT.0.0) GO TO 1000
0790      IF(RAD.GT.0.0) GO TO 90
0791      RE11 = -A2/(2.*A1)
0792      S1 = 1. - 7.*RE11**2
0793      R13 = (S1-1.)/( -7.)
0794      C
0795      C*** ASSIGN APPROPRIATE VALUE OF FB ACCORDING TO TABLE 6 IN REPORT 2923
0796      C
0797      IF(B.LT.C3) GO TO 27
0798      GO TO 28
0799      C
0800      C*** FIND RSGIC FROM EQUATION 1 IN REPORT 2923 FOR STABLE CONDITIONS
0801      C
0802      27      FB3 = 3.4 - 725.5*(-C2 + B)
0803      RSGIC = (E*FB3)/(Z020 - 7.*R13/(1. - 7.*R13))
0804      SIGR20 = (E*FB3)/(C6+Z020)
0805      IF(R13.GE.C5) RSGIC = SIGR20
0806      GO TO 1000
0807      28      FB4 = 1.55 + 38.04*(B - .0008)
0808      FB5 = 2.35 + 5.43*(B - C7)
0809      RSGIC = (E*FB4)/(Z020 - 7.*R13/(1. - 7.*R13))
0810      IF(B.GE.C7)RSGIC = (E*FB5)/(Z020 - 7.*R13/(1. - 7.*R13))
0811      SIGR21 = (E*FB4)/(C6 + Z020)
0812      SIGR22 = (E*FB5)/(C6 + Z020)
0813      IF(R13.GE.C5.AND.B.LT.C7)RSGIC = SIGR21
0814      IF(R13.GE.C5.AND.B.GE.C7)RSGIC = SIGR22
0815      GO TO 1000
0816      90      RE1 = (-A2 + 80RT(RAD))/(2.*A1)
0817      R14 = RE1**2
0818      C
0819      C*** ASSIGN PROPER VALUE OF FB ACCORDING TO TABLE 6 IN REPORT 2923
0820      C
0821      IF(B.LT.C3) GO TO 37
0822      GO TO 38
0823      C
0824      C*** FIND THE RICHARDSON NUMBER USING EQUATION 2(A) FOR UNSTABLE
0825      C*** CONDITIONS
0826      C
0827      37      FB3 = 3.4 - 725.5*(-C2+B)
0828      RSGIC = (E*FB3)/(Z020 - 7.*R14/(1. - 7.*R14))
0829      SIGR20 = (E*FB3)/(C6+Z020)
0830      IF(R14.GE.C5) RSGIC = SIGR20
0831      GO TO 1000
0832      38      FB4 = 1.55 + 38.04*(B - .0008)
0833      FB5 = 2.35 + 5.43*(B - C7)
0834      RSGIC = (E*FB4)/(Z020 - 7.*R14/(1. - 7.*R14))
0835      IF(B.GE.C7)RSGIC = (E*FB5)/(Z020 - 7.*R14/(1. - 7.*R14))
0836      SIGR21 = (E*FB4)/(C6+Z020)
0837      SIGR22 = (E*FB5)/(C6+Z020)
0838      IF(R14.GE.C5.AND.B.LT.C7)RSGIC = SIGR21
0839      IF(R14.GE.C5.AND.B.GE.C7)RSGIC = SIGR22

```



ORIGINAL PAGE IS  
OF POOR QUALITY

```
0039      IF(R14.GE.C5.AND.B.GE.C7) RSIG = S1GR22  
0040      C  
0041      C*** CHECK FOR VALI) S1GA VALUE  
0042      C  
0043      I000 IF (RSIG.LE.0. OR. RSIG.GT.30.) RSIG = 30.  
0044      RETURN  
0045      END  
0046      END0
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```

0001 FTI4
0002 PROGRAM RMTPS(3),REVISED MET PLOT PROG *CAB,810813
0003 C
0004 C
0005 C * THIS PROGRAM GENERATES A METEOROLOGICAL PROFILE OF A SOUNDING
0006 C * UPON THE FLASMASCOPE WITH THE EFFECTIVE CLOUD HEIGHT DRAWN
0007 C * AS AN ELLIPSE JUST BELOW THE TOP OF THE SURFACE LAYER.
0008 C
0009 C
0010 C >>> COMMON AREAS
0011 C >>> MATH PARAMETERS
0012 C COMMON PI,PIOVK2,P143,TWOPI,SQR2PI,DEGRAD,RAD2EG,IV2
0013 C
0014 C >>> VEHICLE PARAMETERS
0015 C COMMON VPAR(10),SIGHCL
0016 C
0017 C >>> OPTION PARAMETERS
0018 C COMMON YPOS,YES,ISINFO,IVHICL,IRUN,ICALC,IPLACE,ITIMEZ,IGOTMP,
0019 C NUMRUN,LU,NUM,MONLY,MFORM,LUPRNT,ITDU,IPL1,IPL2,IPL3
0020 C INTEGER YES
0021 C
0022 C >>> MET DATA
0023 C COMMON ALT(30),IDIR(30),SPEED(30),TEMP(30),PRESS(30),PTEMP(30),
0024 C SURDEN,FILE(3)
0025 C INTEGER FILE
0026 C
0027 C >>> CALCULATION TIME
0028 C COMMON ITIME,IDAY,IMONTH(2),IYEAR
0029 C
0030 C >>> SOUNDING TIME
0031 C COMMON ISTEME,ISDAY,ISMONK(2),ISYEAR
0032 C
0033 C >>> LAUNCH TIME
0034 C COMMON LTIME,LTIM,LDAY,LMONTH(2),LYEAR,LAUNTD(10)
0035 C LOGICAL LTIME,LYRCLT,LYRPLB
0036 C
0037 C >>> CLOUD STABILIZATION AND LAYER PARAMETERS
0038 C COMMON STBLT,STBHT,STBAZD,STBRNG,STBTIM,CLDRAD,RISTIM(30),BOTLAY,
0039 C TOPLAY,CALHT,LAYTOP,LAYBOT,REFLEC,IPTZ
0040 C
0041 C >>> DIFFUSION COEFFICIENTS
0042 C COMMON SIGX,SIGY,SIGZ0,SIGAP,XDIST,YDIST,EXPZ,SZSICZ,SIGY,SIGAL,
0043 C CLDLG,AVSPD,AVDIR,SIGAZ
0044 C
0045 C >>> CONCENTRATION AND DOSAGE VALUES AND RANGES
0046 C COMMON CONMAX,CONCPK,DOSMAX,DOSFX,RNGSNC,RNGSMD
0047 C
0048 C >>> EQUIVALENCE STATEMENT FOR VEHICLE PARAMETERS
0049 C EQUIVALENCE (CQ1,VPAR(1)),(CQ2,VPAR(2)),(CQ3,VPAR(3)),
0050 C (CQ4,VPAR(4)),(CQ5,VPAR(5)),(CQ6,VPAR(6)),
0051 C (CQ7,VPAR(7)),(CQ8,VPAR(8)),(CQ9,VPAR(9)),
0052 C (CQ10,VPAR(10)),(CQ11,VPAR(11)),(CQ12,VPAR(12)),
0053 C (CQ13,VPAR(13)),(CQ14,VPAR(14)),(CQ15,VPAR(15)),
0054 C (CQ16,VPAR(16)),(CQ17,VPAR(17)),(CQ18,VPAR(18))
0055 C
0056 C
0057 C
0058 C
0059 C
0060 C
0061 C
0062 C
0063 C
0064 C
0065 C
0066 C
0067 C
0068 C
0069 C
0070 C
0071 C
0072 C
0073 C
0074 C
0075 C
0076 C
0077 C
0078 C
0079 C
0080 C
0081 C
0082 C
0083 C
0084 C
0085 C
0086 C
0087 C
0088 C
0089 C
0090 C
0091 C
0092 C
0093 C
0094 C
0095 C
0096 C
0097 C
0098 C
0099 C
0100 C
0101 C
0102 C
0103 C
0104 C
0105 C
0106 C
0107 C
0108 C
0109 C
0110 C
0111 C
0112 C
0113 C
0114 C
0115 C
0116 C
0117 C
0118 C
0119 C
0120 C
0121 C
0122 C
0123 C
0124 C
0125 C
0126 C
0127 C
0128 C
0129 C
0130 C
0131 C
0132 C
0133 C
0134 C
0135 C
0136 C
0137 C
0138 C
0139 C
0140 C
0141 C
0142 C
0143 C
0144 C
0145 C
0146 C
0147 C
0148 C
0149 C
0150 C
0151 C
0152 C
0153 C
0154 C
0155 C
0156 C
0157 C
0158 C
0159 C
0160 C
0161 C
0162 C
0163 C
0164 C
0165 C
0166 C
0167 C
0168 C
0169 C
0170 C
0171 C
0172 C
0173 C
0174 C
0175 C
0176 C
0177 C
0178 C
0179 C
0180 C
0181 C
0182 C
0183 C
0184 C
0185 C
0186 C
0187 C
0188 C
0189 C
0190 C
0191 C
0192 C
0193 C
0194 C
0195 C
0196 C
0197 C
0198 C
0199 C
0200 C
0201 C
0202 C
0203 C
0204 C
0205 C
0206 C
0207 C
0208 C
0209 C
0210 C
0211 C
0212 C
0213 C
0214 C
0215 C
0216 C
0217 C
0218 C
0219 C
0220 C
0221 C
0222 C
0223 C
0224 C
0225 C
0226 C
0227 C
0228 C
0229 C
0230 C
0231 C
0232 C
0233 C
0234 C
0235 C
0236 C
0237 C
0238 C
0239 C
0240 C
0241 C
0242 C
0243 C
0244 C
0245 C
0246 C
0247 C
0248 C
0249 C
0250 C
0251 C
0252 C
0253 C
0254 C
0255 C
0256 C
0257 C
0258 C
0259 C
0260 C
0261 C
0262 C
0263 C
0264 C
0265 C
0266 C
0267 C
0268 C
0269 C
0270 C
0271 C
0272 C
0273 C
0274 C
0275 C
0276 C
0277 C
0278 C
0279 C
0280 C
0281 C
0282 C
0283 C
0284 C
0285 C
0286 C
0287 C
0288 C
0289 C
0290 C
0291 C
0292 C
0293 C
0294 C
0295 C
0296 C
0297 C
0298 C
0299 C
0300 C
0301 C
0302 C
0303 C
0304 C
0305 C
0306 C
0307 C
0308 C
0309 C
0310 C
0311 C
0312 C
0313 C
0314 C
0315 C
0316 C
0317 C
0318 C
0319 C
0320 C
0321 C
0322 C
0323 C
0324 C
0325 C
0326 C
0327 C
0328 C
0329 C
0330 C
0331 C
0332 C
0333 C
0334 C
0335 C
0336 C
0337 C
0338 C
0339 C
0340 C
0341 C
0342 C
0343 C
0344 C
0345 C
0346 C
0347 C
0348 C
0349 C
0350 C
0351 C
0352 C
0353 C
0354 C
0355 C
0356 C
0357 C
0358 C
0359 C
0360 C
0361 C
0362 C
0363 C
0364 C
0365 C
0366 C
0367 C
0368 C
0369 C
0370 C
0371 C
0372 C
0373 C
0374 C
0375 C
0376 C
0377 C
0378 C
0379 C
0380 C
0381 C
0382 C
0383 C
0384 C
0385 C
0386 C
0387 C
0388 C
0389 C
0390 C
0391 C
0392 C
0393 C
0394 C
0395 C
0396 C
0397 C
0398 C
0399 C
0400 C
0401 C
0402 C
0403 C
0404 C
0405 C
0406 C
0407 C
0408 C
0409 C
0410 C
0411 C
0412 C
0413 C
0414 C
0415 C
0416 C
0417 C
0418 C
0419 C
0420 C
0421 C
0422 C
0423 C
0424 C
0425 C
0426 C
0427 C
0428 C
0429 C
0430 C
0431 C
0432 C
0433 C
0434 C
0435 C
0436 C
0437 C
0438 C
0439 C
0440 C
0441 C
0442 C
0443 C
0444 C
0445 C
0446 C
0447 C
0448 C
0449 C
0450 C
0451 C
0452 C
0453 C
0454 C
0455 C
0456 C
0457 C
0458 C
0459 C
0460 C
0461 C
0462 C
0463 C
0464 C
0465 C
0466 C
0467 C
0468 C
0469 C
0470 C
0471 C
0472 C
0473 C
0474 C
0475 C
0476 C
0477 C
0478 C
0479 C
0480 C
0481 C
0482 C
0483 C
0484 C
0485 C
0486 C
0487 C
0488 C
0489 C
0490 C
0491 C
0492 C
0493 C
0494 C
0495 C
0496 C
0497 C
0498 C
0499 C
0500 C
0501 C
0502 C
0503 C
0504 C
0505 C
0506 C
0507 C
0508 C
0509 C
0510 C
0511 C
0512 C
0513 C
0514 C
0515 C
0516 C
0517 C
0518 C
0519 C
0520 C
0521 C
0522 C
0523 C
0524 C
0525 C
0526 C
0527 C
0528 C
0529 C
0530 C
0531 C
0532 C
0533 C
0534 C
0535 C
0536 C
0537 C
0538 C
0539 C
0540 C
0541 C
0542 C
0543 C
0544 C
0545 C
0546 C
0547 C
0548 C
0549 C
0550 C
0551 C
0552 C
0553 C
0554 C
0555 C
0556 C
0557 C
0558 C
0559 C
0560 C
0561 C
0562 C
0563 C
0564 C
0565 C
0566 C
0567 C
0568 C
0569 C
0570 C
0571 C
0572 C
0573 C
0574 C
0575 C
0576 C
0577 C
0578 C
0579 C
0580 C
0581 C
0582 C
0583 C
0584 C
0585 C
0586 C
0587 C
0588 C
0589 C
0590 C
0591 C
0592 C
0593 C
0594 C
0595 C
0596 C
0597 C
0598 C
0599 C
0600 C
0601 C
0602 C
0603 C
0604 C
0605 C
0606 C
0607 C
0608 C
0609 C
0610 C
0611 C
0612 C
0613 C
0614 C
0615 C
0616 C
0617 C
0618 C
0619 C
0620 C
0621 C
0622 C
0623 C
0624 C
0625 C
0626 C
0627 C
0628 C
0629 C
0630 C
0631 C
0632 C
0633 C
0634 C
0635 C
0636 C
0637 C
0638 C
0639 C
0640 C
0641 C
0642 C
0643 C
0644 C
0645 C
0646 C
0647 C
0648 C
0649 C
0650 C
0651 C
0652 C
0653 C
0654 C
0655 C
0656 C
0657 C
0658 C
0659 C
0660 C
0661 C
0662 C
0663 C
0664 C
0665 C
0666 C
0667 C
0668 C
0669 C
0670 C
0671 C
0672 C
0673 C
0674 C
0675 C
0676 C
0677 C
0678 C
0679 C
0680 C
0681 C
0682 C
0683 C
0684 C
0685 C
0686 C
0687 C
0688 C
0689 C
0690 C
0691 C
0692 C
0693 C
0694 C
0695 C
0696 C
0697 C
0698 C
0699 C
0700 C
0701 C
0702 C
0703 C
0704 C
0705 C
0706 C
0707 C
0708 C
0709 C
0710 C
0711 C
0712 C
0713 C
0714 C
0715 C
0716 C
0717 C
0718 C
0719 C
0720 C
0721 C
0722 C
0723 C
0724 C
0725 C
0726 C
0727 C
0728 C
0729 C
0730 C
0731 C
0732 C
0733 C
0734 C
0735 C
0736 C
0737 C
0738 C
0739 C
0740 C
0741 C
0742 C
0743 C
0744 C
0745 C
0746 C
0747 C
0748 C
0749 C
0750 C
0751 C
0752 C
0753 C
0754 C
0755 C
0756 C
0757 C
0758 C
0759 C
0760 C
0761 C
0762 C
0763 C
0764 C
0765 C
0766 C
0767 C
0768 C
0769 C
0770 C
0771 C
0772 C
0773 C
0774 C
0775 C
0776 C
0777 C
0778 C
0779 C
0780 C
0781 C
0782 C
0783 C
0784 C
0785 C
0786 C
0787 C
0788 C
0789 C
0790 C
0791 C
0792 C
0793 C
0794 C
0795 C
0796 C
0797 C
0798 C
0799 C
0800 C
0801 C
0802 C
0803 C
0804 C
0805 C
0806 C
0807 C
0808 C
0809 C
0810 C
0811 C
0812 C
0813 C
0814 C
0815 C
0816 C
0817 C
0818 C
0819 C
0820 C
0821 C
0822 C
0823 C
0824 C
0825 C
0826 C
0827 C
0828 C
0829 C
0830 C
0831 C
0832 C
0833 C
0834 C
0835 C
0836 C
0837 C
0838 C
0839 C
0840 C
0841 C
0842 C
0843 C
0844 C
0845 C
0846 C
0847 C
0848 C
0849 C
0850 C
0851 C
0852 C
0853 C
0854 C
0855 C
0856 C
0857 C
0858 C
0859 C
0860 C
0861 C
0862 C
0863 C
0864 C
0865 C
0866 C
0867 C
0868 C
0869 C
0870 C
0871 C
0872 C
0873 C
0874 C
0875 C
0876 C
0877 C
0878 C
0879 C
0880 C
0881 C
0882 C
0883 C
0884 C
0885 C
0886 C
0887 C
0888 C
0889 C
0890 C
0891 C
0892 C
0893 C
0894 C
0895 C
0896 C
0897 C
0898 C
0899 C
0900 C
0901 C
0902 C
0903 C
0904 C
0905 C
0906 C
0907 C
0908 C
0909 C
0910 C
0911 C
0912 C
0913 C
0914 C
0915 C
0916 C
0917 C
0918 C
0919 C
0920 C
0921 C
0922 C
0923 C
0924 C
0925 C
0926 C
0927 C
0928 C
0929 C
0930 C
0931 C
0932 C
0933 C
0934 C
0935 C
0936 C
0937 C
0938 C
0939 C
0940 C
0941 C
0942 C
0943 C
0944 C
0945 C
0946 C
0947 C
0948 C
0949 C
0950 C
0951 C
0952 C
0953 C
0954 C
0955 C
0956 C
0957 C
0958 C
0959 C
0960 C
0961 C
0962 C
0963 C
0964 C
0965 C
0966 C
0967 C
0968 C
0969 C
0970 C
0971 C
0972 C
0973 C
0974 C
0975 C
0976 C
0977 C
0978 C
0979 C
0980 C
0981 C
0982 C
0983 C
0984 C
0985 C
0986 C
0987 C
0988 C
0989 C
0990 C
0991 C
0992 C
0993 C
0994 C
0995 C
0996 C
0997 C
0998 C
0999 C
1000 C

```



ORIGINAL PAGE IS  
OF POOR QUALITY

```

0119 2HVE,2HR /
0120 DATA ISURTI/2HSU,2HRF,2HAC,2HE,2H,2HTO,2HP,2HLA,2HVE,2HR,
0121 2N,2HSD,2HT,2HLA,2HVE,2HR /
0122 DATA IDATL/2HQA,2HTE/,ITIML/2HTI,2HME/
0123 DATA IALIL/2H A,2H L,2H T,2H I,2H U,2H O,2H E/
0124 DATA TSURX/100,130,140,170,180,210,220,250,260,250,
1300,330,340,370,380,410,420,450,460,490,
0125 DATA BSURX/100,0,160,0,182,5,242,5,265,0,325,0,347,5,407,5,430,0,
0126 490,0/
0127 DATA ITOP/2H T,2HOP/,1BOT/2H B,2HOT/
0128 DATA IKNUM/2H10,2H-5,2H 0,2H 5,2H10,2H15,2H20,2H25,2H30,2H35,
0129 2H40,2H45,2H50/
0130 DATA XAX/460,0,100,0,100,0/,YAX/90,0,90,0,410,0/
0131 DATA YCO/90,90,410,410/
0132 DATA IYNUM/2H 2H 0,2H 4,2H00,2H 0,2H00,2H12,2H00,2H16,2H00,
0133 12H20,2H00,2H24,2H00,2H28,2H00,2H32,2H00,2H36,2H00,2H40,2H00,
0134 DATA IDIH/2H10,2H00,2H12,2H00,2H 0,2H00,2H 4,2H00,2H 2H0 ,
0135 2H 4,2H00,2H 0,2H00,2H12,2H00,2H16,2H00/
0136 DATA ICWD/2HCR,2HOS,2HS,2HUI,2HND,2H 0,2HIN,2H (.2HM)/
0137 DATA IMET/2H(M,IN)/,INSTAL/2HVA,2HFB,2HKS,INC/
0138 DATA ISTAB/2HST,2HAB,2H H,2HTI/,ISICHA/2HSI,2HGM,2HA1/
0139 DATA IFOUR/2H40,2H00/
0140 DATA NAFTI/2HFI,2HLE,2H U,2HSE,2HD1/
0141 C
0142 C
0143 C
0144 C
0145 C
0146 C
0147 C
0148 C
0149 C
0150 C
0151 C
0152 C
0153 C
0154 C
0155 C
0156 C
0157 C
0158 C
0159 C
0160 C
0161 C
0162 C
0163 C
0164 C
0165 C
0166 C
0167 C
0168 C
0169 C
0170 C
0171 C
0172 C
0173 C
0174 C
0175 C
0176 C
0177 C
0178 C

```

THE FOLLOWING DATA STATEMENT CONTAINS OCTAL REPRESENTATION  
OF AN ALTERNATE CHARACTER SET (5 BY 6 RASTER UNITS IN SIZE)  
AS FOLLOWS: 0-9,A-Z, AND SPECIAL CHARACTERS +,-,\*,/,(<,>)

```

DATA IALTC/368,418,418,368,400,0,218,778,18,400,
0148 238,458,458,318,400,428,418,518,668,400,
0149 148,248,778,48,400,728,518,518,468,400,
0150 368,458,458,400,608,438,448,708,400,
0151 268,518,518,268,400,208,518,518,368,400,
0152 378,508,508,378,400,778,518,518,268,400,
0153 368,418,418, 228,400,778,418,418,368,400,
0154 778,518,518,418,400,778,508,508,408,400,
0155 368,418,458,268,400,778,108,108,778,400,
0156 0,418,778,418,400,428,418,768,408,400,
0157 778,148, 228,418,400,778,18,18,400,
0158 778,208,108,208,778,300,778,308,68,778,400,
0159 368,418,418,368,400,778,448,448,308,400,
0160 348,428,428,358,400,778,448,468,318,400,
0161 228,518,458,228,400,408,408,778,408,408,300,
0162 768,18,18,768,400,748,28,18,28,748,300,
0163 768,18,368,18,768,300,618,128,048,128,618,300,
0164 608,108,178,108,608,300,418,438,458,518,618,300,
0165 2048,378,2048,300,508,300,218,128,378,128,218,300,
0166 18,28,46,108,208,300,0,368,418,500,0,418,368,500/
0167 DATA IALTC1/0,128,128,128,400/
0168 DATA IALTF/0,18,600/
0169 DATA IALTC2/0,128,600/
0170 DATA IALTS/800/
0171 C
0172 C
0173 C
0174 C
0175 C
0176 C
0177 C
0178 C

```

READ THE COMMON DISK FILE  
CALL RUDIS(NAME,0)

CALL SUBROUTINE GRAF TO INITIALIZE GRAPHIC MODE ON PLASMASCOPE  
END (Graf, 1)

0,1  
2,3  
4,5  
6,7  
8,9  
A,B  
C,D  
E,F  
G,H  
I,J  
K,L  
M,N  
O,P  
Q,R  
S,T  
U,V  
W,X  
Y,Z  
+,-  
\*,/(<,>  
=,SP

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0179 C
0180 C CALL SUBROUTINE VERSH TO DETERMINE IF RUNNING ON PLASMASCOPE
0181 C (IVERSH=0) OR CRT(IVERSH=1)
0182 C
0183 C CALL VERSH(IVERSH)
0184 C LVRCRT = IVERSH .NE. 0
0185 C LVAPLS = .NOT.LVRCRT
0186 C
0187 C CALL SUBROUTINE CLEAR TO CLEAR SCREEN IF RUNNING ON THE PLASMASCOPE
0188 C
0189 C IF(LVAPLS) CALL CLEAR
0190 C
0191 C LOAD THE ALTERNATE CHARACTER SET INTO THE PLASMASCOPE
0192 C
0193 C CALL LALT(INH,IALTCH,10)
0194 C CALL LALT(INH,IALTCH(81),26)
0195 C CALL LALT(INH,IALTCH(289),6)
0196 C CALL LALT(INH,IALTCH(1),1)
0197 C CALL LALT(INH,IALTSP,1)
0198 C CALL LALT(INH,IALTCL,1)
0199 C CALL LALT(INH,IALTP,1)
0200 C
0201 C .....
0202 C
0203 C THIS SECTION DRAWS THE NET PLOT FORM
0204 C
0205 C .....
0206 C
0207 C XSCALE = 0.6
0208 C
0209 C *** IF CRT VERSION IN AND FORM DRAWING NOT DESIRED, DELETE FORM DRAWING
0210 C IF( LVRCRT .AND. (IFORM.EQ.0) ) GO TO 34
0211 C
0212 C DRAW THE DATE LABEL
0213 C
0214 C
0215 C CALL CHAR(20,0,490,0,0,1DATL,4,2,1)
0216 C XL(1) = 20.0
0217 C XL(2) = 48.0
0218 C YL(1) = 400.0
0219 C YL(2) = 408.0
0220 C CALL LINE(XL,YL,2,0)
0221 C
0222 C DRAW THE TIME LABEL
0223 C
0224 C CALL CHAR(164,0,490,0,0,ITIML,4,2,1)
0225 C XL(1) = 164.0
0226 C XL(2) = 192.0
0227 C CALL LINE(XL,YL,2,0)
0228 C
0229 C DRAW THE SURFACE PRESSURE, DENSITY, AND, IF REQUIRED,
0230 C STABILIZATION HEIGHT AND SIGMA LABELS
0231 C
0232 C CALL CHAR(20,0,475,0,0,ISURLI,6,2,1)
0233 C IF(LVAPLS)CALL CHAR(468,0,478,0,0,IEXP3,1,2,1)
0234 C IF(LVRCRT)CALL CHAR(316,0,478,0,0,IEXP3,1,2,1)
0235 C XL(1) = 20.0
0236 C XL(2) = 76.0
0237 C YL(1) = 473.0
0238 C YL(2) = 473.0
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0239 CALL LINE(XL, YL, 2, 0)
0240 IFC(LVRFPLS)GO TO 3
0241 CALL CHAR(374, 0, 475, 0, 0, ISTAB, 9, 2, 1)
0242 CALL CHAR(466, 0, 475, 0, 0, ISTL(4), 1, 2, 1)
0243 XL(1) = 374.0
0244 XL(2) = 422.0
0245 CALL LINE(XL, YL, 2, 0)
0246 CALL CHAR(374, 0, 490, 0, 0, ISIGMA, 6, 2, 1)
0247 XL(1) = 410.0
0248 XL(2) = 488.0
0249 YL(1) = 488.0
0250 CALL LINE(XL, YL, 2, 0)
0251 C
0252 C
0253 C
0254 C
0255 C
0256 C
0257 C
0258 C
0259 C
0260 C
0261 C
0262 C
0263 C
0264 C
0265 C
0266 C
0267 C
0268 C
0269 C
0270 C
0271 C
0272 C
0273 C
0274 C
0275 C
0276 C
0277 C
0278 C
0279 C
0280 C
0281 C
0282 C
0283 C
0284 C
0285 C
0286 C
0287 C
0288 C
0289 C
0290 C
0291 C
0292 C
0293 C
0294 C
0295 C
0296 C
0297 C
0298 C
0299 C
0300 C
0301 C
0302 C
0303 C
0304 C
0305 C
0306 C
0307 C
0308 C
0309 C
0310 C
0311 C
0312 C
0313 C
0314 C
0315 C
0316 C
0317 C
0318 C
0319 C
0320 C
0321 C
0322 C
0323 C
0324 C
0325 C
0326 C
0327 C
0328 C
0329 C
0330 C
0331 C
0332 C
0333 C
0334 C
0335 C
0336 C
0337 C
0338 C
0339 C
0340 C
0341 C
0342 C
0343 C
0344 C
0345 C
0346 C
0347 C
0348 C
0349 C
0350 C
0351 C
0352 C
0353 C
0354 C
0355 C
0356 C
0357 C
0358 C
0359 C
0360 C
0361 C
0362 C
0363 C
0364 C
0365 C
0366 C
0367 C
0368 C
0369 C
0370 C
0371 C
0372 C
0373 C
0374 C
0375 C
0376 C
0377 C
0378 C
0379 C
0380 C
0381 C
0382 C
0383 C
0384 C
0385 C
0386 C
0387 C
0388 C
0389 C
0390 C
0391 C
0392 C
0393 C
0394 C
0395 C
0396 C
0397 C
0398 C
0399 C
0400 C
0401 C
0402 C
0403 C
0404 C
0405 C
0406 C
0407 C
0408 C
0409 C
0410 C
0411 C
0412 C
0413 C
0414 C
0415 C
0416 C
0417 C
0418 C
0419 C
0420 C
0421 C
0422 C
0423 C
0424 C
0425 C
0426 C
0427 C
0428 C
0429 C
0430 C
0431 C
0432 C
0433 C
0434 C
0435 C
0436 C
0437 C
0438 C
0439 C
0440 C
0441 C
0442 C
0443 C
0444 C
0445 C
0446 C
0447 C
0448 C
0449 C
0450 C
0451 C
0452 C
0453 C
0454 C
0455 C
0456 C
0457 C
0458 C
0459 C
0460 C
0461 C
0462 C
0463 C
0464 C
0465 C
0466 C
0467 C
0468 C
0469 C
0470 C
0471 C
0472 C
0473 C
0474 C
0475 C
0476 C
0477 C
0478 C
0479 C
0480 C
0481 C
0482 C
0483 C
0484 C
0485 C
0486 C
0487 C
0488 C
0489 C
0490 C
0491 C
0492 C
0493 C
0494 C
0495 C
0496 C
0497 C
0498 C
0499 C
0500 C
0501 C
0502 C
0503 C
0504 C
0505 C
0506 C
0507 C
0508 C
0509 C
0510 C
0511 C
0512 C
0513 C
0514 C
0515 C
0516 C
0517 C
0518 C
0519 C
0520 C
0521 C
0522 C
0523 C
0524 C
0525 C
0526 C
0527 C
0528 C
0529 C
0530 C
0531 C
0532 C
0533 C
0534 C
0535 C
0536 C
0537 C
0538 C
0539 C
0540 C
0541 C
0542 C
0543 C
0544 C
0545 C
0546 C
0547 C
0548 C
0549 C
0550 C
0551 C
0552 C
0553 C
0554 C
0555 C
0556 C
0557 C
0558 C
0559 C
0560 C
0561 C
0562 C
0563 C
0564 C
0565 C
0566 C
0567 C
0568 C
0569 C
0570 C
0571 C
0572 C
0573 C
0574 C
0575 C
0576 C
0577 C
0578 C
0579 C
0580 C
0581 C
0582 C
0583 C
0584 C
0585 C
0586 C
0587 C
0588 C
0589 C
0590 C
0591 C
0592 C
0593 C
0594 C
0595 C
0596 C
0597 C
0598 C
0599 C
0600 C
0601 C
0602 C
0603 C
0604 C
0605 C
0606 C
0607 C
0608 C
0609 C
0610 C
0611 C
0612 C
0613 C
0614 C
0615 C
0616 C
0617 C
0618 C
0619 C
0620 C
0621 C
0622 C
0623 C
0624 C
0625 C
0626 C
0627 C
0628 C
0629 C
0630 C
0631 C
0632 C
0633 C
0634 C
0635 C
0636 C
0637 C
0638 C
0639 C
0640 C
0641 C
0642 C
0643 C
0644 C
0645 C
0646 C
0647 C
0648 C
0649 C
0650 C
0651 C
0652 C
0653 C
0654 C
0655 C
0656 C
0657 C
0658 C
0659 C
0660 C
0661 C
0662 C
0663 C
0664 C
0665 C
0666 C
0667 C
0668 C
0669 C
0670 C
0671 C
0672 C
0673 C
0674 C
0675 C
0676 C
0677 C
0678 C
0679 C
0680 C
0681 C
0682 C
0683 C
0684 C
0685 C
0686 C
0687 C
0688 C
0689 C
0690 C
0691 C
0692 C
0693 C
0694 C
0695 C
0696 C
0697 C
0698 C
0699 C
0700 C
0701 C
0702 C
0703 C
0704 C
0705 C
0706 C
0707 C
0708 C
0709 C
0710 C
0711 C
0712 C
0713 C
0714 C
0715 C
0716 C
0717 C
0718 C
0719 C
0720 C
0721 C
0722 C
0723 C
0724 C
0725 C
0726 C
0727 C
0728 C
0729 C
0730 C
0731 C
0732 C
0733 C
0734 C
0735 C
0736 C
0737 C
0738 C
0739 C
0740 C
0741 C
0742 C
0743 C
0744 C
0745 C
0746 C
0747 C
0748 C
0749 C
0750 C
0751 C
0752 C
0753 C
0754 C
0755 C
0756 C
0757 C
0758 C
0759 C
0760 C
0761 C
0762 C
0763 C
0764 C
0765 C
0766 C
0767 C
0768 C
0769 C
0770 C
0771 C
0772 C
0773 C
0774 C
0775 C
0776 C
0777 C
0778 C
0779 C
0780 C
0781 C
0782 C
0783 C
0784 C
0785 C
0786 C
0787 C
0788 C
0789 C
0790 C
0791 C
0792 C
0793 C
0794 C
0795 C
0796 C
0797 C
0798 C
0799 C
0800 C
0801 C
0802 C
0803 C
0804 C
0805 C
0806 C
0807 C
0808 C
0809 C
0810 C
0811 C
0812 C
0813 C
0814 C
0815 C
0816 C
0817 C
0818 C
0819 C
0820 C
0821 C
0822 C
0823 C
0824 C
0825 C
0826 C
0827 C
0828 C
0829 C
0830 C
0831 C
0832 C
0833 C
0834 C
0835 C
0836 C
0837 C
0838 C
0839 C
0840 C
0841 C
0842 C
0843 C
0844 C
0845 C
0846 C
0847 C
0848 C
0849 C
0850 C
0851 C
0852 C
0853 C
0854 C
0855 C
0856 C
0857 C
0858 C
0859 C
0860 C
0861 C
0862 C
0863 C
0864 C
0865 C
0866 C
0867 C
0868 C
0869 C
0870 C
0871 C
0872 C
0873 C
0874 C
0875 C
0876 C
0877 C
0878 C
0879 C
0880 C
0881 C
0882 C
0883 C
0884 C
0885 C
0886 C
0887 C
0888 C
0889 C
0890 C
0891 C
0892 C
0893 C
0894 C
0895 C
0896 C
0897 C
0898 C
0899 C
0900 C
0901 C
0902 C
0903 C
0904 C
0905 C
0906 C
0907 C
0908 C
0909 C
0910 C
0911 C
0912 C
0913 C
0914 C
0915 C
0916 C
0917 C
0918 C
0919 C
0920 C
0921 C
0922 C
0923 C
0924 C
0925 C
0926 C
0927 C
0928 C
0929 C
0930 C
0931 C
0932 C
0933 C
0934 C
0935 C
0936 C
0937 C
0938 C
0939 C
0940 C
0941 C
0942 C
0943 C
0944 C
0945 C
0946 C
0947 C
0948 C
0949 C
0950 C
0951 C
0952 C
0953 C
0954 C
0955 C
0956 C
0957 C
0958 C
0959 C
0960 C
0961 C
0962 C
0963 C
0964 C
0965 C
0966 C
0967 C
0968 C
0969 C
0970 C
0971 C
0972 C
0973 C
0974 C
0975 C
0976 C
0977 C
0978 C
0979 C
0980 C
0981 C
0982 C
0983 C
0984 C
0985 C
0986 C
0987 C
0988 C
0989 C
0990 C
0991 C
0992 C
0993 C
0994 C
0995 C
0996 C
0997 C
0998 C
0999 C
1000 C
```

CALL LINE(XL, YL, 2, 0)  
IF(LVRFPLS)GO TO 3  
CALL CHAR(374, 0, 475, 0, 0, ISTAB, 9, 2, 1)  
CALL CHAR(466, 0, 475, 0, 0, ISTL(4), 1, 2, 1)  
XL(1) = 374.0  
XL(2) = 422.0  
CALL LINE(XL, YL, 2, 0)  
CALL CHAR(374, 0, 490, 0, 0, ISIGMA, 6, 2, 1)  
XL(1) = 410.0  
YL(1) = 488.0  
YL(2) = 488.0  
CALL LINE(XL, YL, 2, 0)  
PRINT SURFACE HEADER -- TOP LAYER HEADER -- BOT LAYER  
HEADER (IF REQUIRED)  
3 IFC(LVRCRT)GO TO 4  
I = 20  
IF(CALC .EQ. 1)I = 32  
CALL CHAR(222, 0, 461, 0, 0, ISURT1, 1, 2, 1)  
GO TO 5  
4 I = 26  
IF(CALC .EQ. 1)I = 44  
CALL CHAR(222, 0, 461, 0, 0, ISURT, 1, 2, 1)  
XL(1) = 222.0  
XL(2) = 278.0  
YL(1) = 459.0  
YL(2) = 459.0  
CALL LINE(XL, YL, 2, 0)  
XL(1) = 302.0  
XL(2) = 374.0  
CALL LINE(XL, YL, 2, 0)  
IF(CALC .NE. 1)GO TO 8  
XL(1) = 398.0  
XL(2) = 470.0  
CALL LINE(XL, YL, 2, 0)  
DRAW DRY TEMPERATURE LABEL  
8 CALL CHAR(30, 0, 450, 0, 0, IDT, 24, 2, 1)  
DRAW POTENTIAL TEMPERATURE LABEL  
CALL CHAR(30, 0, 440, 0, 0, IPT, 22, 2, 1)  
DRAW WIND SPEED LABEL  
CALL CHAR(30, 0, 430, 0, 0, IUS, 16, 2, 1)  
DRAW WIND DIRECTION LABEL  
CALL CHAR(30, 0, 420, 0, 0, IMD, 20, 2, 1)  
DRAW X AND Y AXIS  
XAX AND YAX ARE SET IN A DATA STATEMENT--THE USE OF THE XSCALE  
C REDUCES THE NET PLOT ON THE X AXIS.

CALL LINE(XAX, YAX, 3, 0)

~~Original  
of Poor~~

~~Original  
of Poor~~

```

0299 C          DRAW X AXIS LABELS
0300 C
0301 C          CALL CHAR(100,0,70,0,0,ISTL,24,2,1)
0302 C
0303 C          DRAW TICK MARKS ON X AXIS
0304 C          XINC IS THE SPACE BETWEEN EACH TIC MARK
0305 C          XINC = (XAX(1) - XAX(2))/12.0
0306 C          TIC = XAX(2) - XINC
0307 C          YL(1) = 88.0
0308 C          YL(2) = 92.0
0309 C          COORD = TIC - 0.0
0310 C          DO 10 I=1,13
0311 C             TIC = TIC + XINC
0312 C             XL(1) = TIC
0313 C             XL(2) = TIC
0314 C          CALL LINE(XL,YL,2,0)
0315 C          XL(1) = XL(1) + XINC/2.0
0316 C          XL(2) = XL(2) + XINC/2.0
0317 C          IF(I.EQ.13)CALL LINE(XL,YL,2,0)
0318 C          COORD = COORD + XINC
0319 C          IF(I.EQ.1)CALL CHAR(84,0,80,0,0,INUM,1,2,1)
0320 C          10 CALL CHAR(COOR,80,0,0,INUM(1),2,2,1)
0321 C
0322 C          DRAW WIND DIRECTION AXIS
0323 C
0324 C          XSCALE IS USED HERE TO SHIFT THE WIND DIRECTION AXIS TO THE LEFT
0325 C          XL(1) = 300.0 * XSCALE
0326 C          XL(2) = 460.0 * XSCALE
0327 C          YL(1) = 60.0
0328 C          YL(2) = 60.0
0329 C          CALL LINE(XL,YL,2,0)
0330 C
0331 C          DRAW WIND DIRECTION AXIS LABEL
0332 C
0333 C          XSCALE IS USED TO SHIFT THE WIND DIRECTION AXIS LABEL TO THE LEFT
0334 C          XC = 310.0 * XSCALE
0335 C          CALL CHAR(XC,40,0,0,IUD,20,2,1)
0336 C
0337 C          DRAW TICK MARKS ON WIND DIRECTION AXIS
0338 C
0339 C          DO 15 K=1,18
0340 C             YL(1) = 58.0
0341 C             YL(2) = 62.0
0342 C             DO 16 I=1,9
0343 C                N = 2 * I - 1
0344 C                16 CALL LINE(XND(N),YL,2,0)
0345 C
0346 C          DRAW TICK MARKS ON Y AXIS
0347 C          TIC = 58.0
0348 C          YL(1) = 98.0
0349 C          YL(2) = 102.0
0350 C
0351 C
0352 C
0353 C
0354 C
0355 C
0356 C
0357 C
0358 C

```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0359 DO 20 I=1,11
0360 TIC = TIC + 32.0
0361 YL(1) = TIC
0362 YL(2) = TIC
0363 N = 2 * I - 1
0364 CALL CHAR(64.0, YL, 0, IYNUM(I), 4.2, 1)
0365 20 CALL LINE(XL, YL, 2, 0)
0366 C
0367 C DRAW Y AXIS LABEL
0368 C
0369 COORD = 360.0
0370 DO 30 I=1,8
0371 COORD = COORD - 20.0
0372 30 CALL CHAR(30.0, COORD, 0, IALTL(I), 2, 2, 1)
0373 CALL CHAR(30.0, COORD-20.0, 0, INET, 3, 2, 1)
0374 C
0375 C THE FOLLOWING THREE SECTIONS OF CODE PRODUCE AN X AND Y AXIS ON
0376 C THE RIGHT HAND SIDE OF THE PLOT
0377 C
0378 C DRAW THE NEW AXES.
0379 C
0380 XCD(1) = 300.0
0381 XCD(2) = 500.0
0382 XCD(3) = 500.0
0383 CALL LINE(XCD, YCD, 3, 0)
0384 C
0385 C DRAW TIC MARKS ON THE RIGHT HAND Y AXIS
0386 C
0387 TIC = 58.0
0388 XL(1) = 498.0
0389 XL(2) = 502.0
0390 DO 76 I = 1,11
0391 TIC = TIC + 32.0
0392 YL(1) = TIC
0393 YL(2) = TIC
0394 76 CALL LINE(XL, YL, 2, 0)
0395 XL = 460.0
0396 YL = 410.0
0397 CALL CHAR(XL, YL, 0, IFOUR, 4.2, 1)
0398 C
0399 C DRAW TICK MARKS ON THE RIGHT HAND X AXIS AND NUMBER THE AXIS
0400 C
0401 TIC = 275.0
0402 YL(1) = 88.0
0403 YL(2) = 92.0
0404 COORD = TIC - 8.0
0405 DO 78 K = 1,9
0406 TIC = TIC + 25.0
0407 XL(1) = TIC
0408 XL(2) = TIC
0409 CALL LINE(XL, YL, 2, 0)
0410 COORD = COORD + 25.0
0411 N = 2 * K - 1
0412 78 CALL CHAR(COORD, 90.0, 0, IDJMK(I), 4.2, 1)
0413 C
0414 C DRAW THE RIGHT HAND X AXIS LABEL
0415 C
0416 CALL CHAR(300.0, 70.0, 0, ICY6, 18, 2, 1)
0417 C
0418 C LABEL FOR THE DATA FILE
```



ORIGINAL PAGE IS  
OF POOR QUALITY

```
0419 C CALL CHAR(404.0,40.0,0,0,MAPIL,10,2,1)
0420 C
0421 C
0422 C
0423 C
0424 C THIS SECTION LABELS THE MET PLOT FORM AND DRAWS THE CURVES
0425 C
0426 C
0427 C
0428 C
0429 C DETERMINE SOME X AND Y COORDINATES AND TOTAL NUMBER OF POINTS
0430 C FOR THE CURVES
0431 C
0432 C 34 CONTINUE
0433 C
0434 C THE FOLLOWING IF TEST ALLOWS MET PLOTS TO BE BY-PASSED
0435 C
0436 C IF((MOH.Y.EQ.1).AND.LVRCRT) GO TO 110
0437 C
0438 C 35 DO 40 I=1,NUM
0439 C IF(ALT(I).GE.4000.0)GO TO 50
0440 C CURVEY(I) = ALI(I) * 0.08 + 90.0
0441 C AMDIR(I) = IDIR(I)
0442 C 40 APTEMP(I) = PTEMP(I) - 273.16
0443 C I = NUM + 1
0444 C 50 ILP = I - 1
0445 C
0446 C CALL SUBROUTINE TO ROTATE WIND DIRECTION FOR PLOTTING
0447 C
0448 C CALL WINDS(AMDIR,ILP,ISC)
0449 C IF(INFORM.NE.0) GO TO 51
0450 C XAX(I) = XAX(I) * XSCALE
0451 C XINC = (XAX(I) - XAX(2))/12.0
0452 C DO 60 K = 1,10
0453 C 80 XUD(K) = XUD(K) * XSCALE
0454 C 51 DO 60 I=1,9
0455 C N = 2 * I - 1
0456 C COORD = XUD(N) - 0.0
0457 C 60 CALL CHAR(COORD,50.0,0,1,MDLN,ISC),4,2,1)
0458 C
0459 C FAKT IS USED TO SCALE THE DATA--EACH XINC REPRESENTS 5 UNITS
0460 C ON THE X AXIS -- 100 IS WHERE THE AXIS BEGINS ---
0461 C
0462 C FAKT = XINC/5.0
0463 C DO 70 I=1,ILP
0464 C USXI) = (SPEED(I) * FAKT) + 2*XINC + 100.0
0465 C DTXI) = (TEMP(I) * FAKT) + 2*XINC + 100.0
0466 C PTXI) = (APTEMP(I) * FAKT) + 2*XINC + 100.0
0467 C IF(TEMP(I).LT.-10.0)DTXI) = 100.0
0468 C IF(TEMP(I).GT.50.0)DTXI) = 460.0 * XSCALE
0469 C IF(APTEMP(I).LT.-10.0)PTXI) = 100.0
0470 C IF(APTEMP(I).GT.50.0)PTXI) = 460.0 * XSCALE
0471 C 70 VDX(I) = (ABS(AMDIR(I)) * 0.444444 + 300.0) * XSCALE
0472 C
0473 C WRITE THE DATE AND TIME
0474 C
0475 C CALL CODE
0476 C WRITE (IALPHA,2000) ISDAY,ISHOR(I),ISHOR(2),ISYEAR
0477 C CALL CHAR(60.,490.,0,IALPHA,11,2,1)
0478 C CALL CODE
0479 C WRITE (IALPHA,2001) ISTIME
```

ORIGINAL PAGE IS  
OF POOR QUALITY

6479 CALL CHAR(204.,490.,0.,IALPHA,4,2,1)  
6480 CALL CHAR(240.,0.,490.,0.,0.,ITIMEZ,1,2,1)  
6481 IF(LVRPLS)CALL CHAR(240.,0.,490.,0.,0.,LAUNITD(4),2,2,1)  
6482 IF(LVRCRT)CALL CHAR(246.,0.,490.,0.,0.,LAUNITD(4),2,2,1)  
6483 C

IF NECESSARY, WRITE THE INSTALLATION

6484 C  
6485 C  
6486 IF(IPLACE.EQ.0)GO TO 74  
6487 I = IPLACE - IPLACE/3  
6488 IF(IPLACE.EQ.4) I = 2  
6489 CALL CHAR(300.,0.,490.,0.,0.,INSTAL(1,1),4,2,1)  
6490 XL(1) = 300.0  
6491 XL(2) = 336.0  
6492 YL(1) = 400.0  
6493 YL(2) = 400.0  
6494 CALL LINE(XL,YL,2.0)  
6495 C

WRITE THE SURFACE PRESSURE, DENSITY, AND STABILIZATION HEIGHT

74 CALL CODE  
6496 C  
6497 C  
6498 WRITE (IALPHA,2002) PRESS(1)  
6499 IF(LVRPLS)CALL CHAR(196.,0.,475.,0.,0.,IALPHA,6,2,1)  
6500 IF(LVRCRT)CALL CHAR(133.,0.,475.,0.,0.,IALPHA,6,2,1)  
6501 CALL CODE  
6502 WRITE (IALPHA,2002) SURDEN  
6503 IF(LVRPLS)CALL CHAR(300.,0.,475.,0.,0.,IALPHA,6,2,1)  
6504 IF(LVRCRT)CALL CHAR(260.,0.,475.,0.,0.,IALPHA,6,2,1)  
6505 IF(LVRPLS)GO TO 77  
6506 CALL CODE  
6507 WRITE (IALPHA,2002) STBALZ  
6508 CALL CHAR(420.,0.,475.,0.,0.,IALPHA,6,2,1)  
6509 CALL CODE  
6510 WRITE (IALPHA,2003) SIGAZ  
6511 CALL CHAR(416.,0.,490.,0.,0.,IALPHA,4,2,1)  
6512 C  
6513 C  
6514 C  
6515 C

WRITE THE DRY TEMPERATURE

77 CALL CODE  
6516 C  
6517 WRITE (IALPHA,2002) TEMP(1)  
6518 CALL CHAR(230.,0.,450.,0.,0.,IALPHA,6,2,1)  
6519 C

WRITE THE POTENTIAL TEMPERATURE

6520 C  
6521 C  
6522 A = PTEMP(1) - 273.16  
6523 CALL CODE  
6524 WRITE (IALPHA,2002) A  
6525 CALL CHAR(230.,0.,440.,0.,0.,IALPHA,6,2,1)  
6526 C

WRITE THE WIND SPEED

6527 C  
6528 C  
6529 CALL CODE  
6530 WRITE (IALPHA,2002) SPEED(1)  
6531 CALL CHAR(230.,0.,430.,0.,0.,IALPHA,6,2,1)  
6532 C

WRITE THE WIND DIRECTION

6533 C  
6534 C  
6535 A = DIR(1)  
6536 CALL CODE  
6537 WRITE (IALPHA,2002) A  
6538 CALL CHAR(230.,0.,420.,0.,0.,IALPHA,6,2,1)  
6539 C

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0539 C          DRAW THE WIND SPEED LINE
0540 C
0541 C
0542 C          CALL DLN<WSK,CURVEY,ILP,0,4,4>
0543 C          CALL DLN<WSK,CURVEY,ILP,0,0,4>
0544 C          COORD = CURVEY<ILP> + 3.0
0545 C          CALL CHAR<WSK<ILP>,COORD,0,ICRVT<1>,2,2,1>
0546 C
0547 C          DRAW THE DRY TEMPERATURE LINE
0548 C
0549 C          CALL LNE<DTX,CURVEY,ILP,0>
0550 C          COORD = CURVEY<ILP> - 5.0
0551 C          CALL CHAR<DTX<ILP>+4.0,COORD,0,ICRVT<2>,2,2,1>
0552 C
0553 C          DRAW THE POTENTIAL TEMPERATURE LINE
0554 C
0555 C          CALL DLN<PTX,CURVEY,ILP,0,4,4>
0556 C          COORD = CURVEY<ILP> + 3.0
0557 C          CALL CHAR<PTX<ILP>,COORD,0,ICRVT<3>,2,2,1>
0558 C
0559 C          DRAW THE WIND DIRECTION LINE
0560 C
0561 C          J1 = 1
0562 C          DO 80 I=2,ILP
0563 C          IF<AND<I>, .GE. 0.0>GO TO 80
0564 C          NUMP = I - 1
0565 C          CALL DLN<WDX<I>,CURVEY<I>,NUMP,0,4,0>
0566 C          CALL LNE<WDX<I>,CURVEY<I>,NUMP,0>
0567 C          J1 = J
0568 C          80 CONTINUE
0569 C          NUMP = ILP - J1 + 1
0570 C          CALL DLN<WDX<I>,CURVEY<I>,NUMP,0,4,0>
0571 C          CALL LNE<WDX<I>,CURVEY<I>,NUMP,0>
0572 C          COORD = CURVEY<ILP> - 5.0
0573 C          CALL CHAR<WDX<ILP>+4.0,COORD,0,ICRVT<4>,2,2,1>
0574 C
0575 C          DRAW TICK MARKS AT THE VALID DATA POINTS ON THE Y AXIS
0576 C
0577 C          XL<1> = 100.0
0578 C          XL<2> = 106.0
0579 C          DO 90 I=1,ILP
0580 C          YL<1> = ALT<I> * 0.08 + 90.0
0581 C          YL<2> = YL<1>
0582 C          90 CALL LNE<XL,YL,2,0>
0583 C
0584 C          DRAW THE CLOUD
0585 C
0586 C          COORD = STRALT * 0.08 + 90.0
0587 C          CALL CLOUD<250.0,COORD>
0588 C          CALL CLOUD<400.0,COORD>
0589 C
0590 C          WRITE OUT THE TOP OF THE SURFACE LAYER LINE AND ALLOW IT
0591 C          TO BE MOVED UP AND DOWN
0592 C
0593 C          CALL MOVEN<LAYTOP,ILP,2,ITOP,310.0,TSURX,10>
0594 C          TOPLAY = ALT<LAYTOP>
0595 C
0596 C          CALCULATE AND WRITE OUT THE EFFECTIVE CLOUD HEIGHT, STRUT
0597 C          SIGK0 = 0.297674 * STRALT
0598 C
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```

0599 STBHT = STBALT
0600 CLDRAD = 2.15 * SIGX0
0601 IV2 = 0
0602 IF<CLDRAD+STBALT .GE. ALT<LAYTOP>>IV2 = 1
0603 SIGZ0 = SIGX0
0604 IF<IV2.EQ.1>SIGZ0 = <ALT<LAYTOP>> - STBALT + <CLDRAD>/4.3
0605 IF<SIGZ0.GT.0.0>GO TO 91
0606 STBHT = 0.5 * ALT<LAYTOP>
0607 SIGZ0 = 0.64 * STBHT/2.15
0608 GO TO 92
0609 91 IF<IV2.EQ.1>STBHT = 0.5 * <ALT<LAYTOP>> + STBALT - CLDRAD
0610 92 WRITE (6,666) STBHT
0611 666 FORMAT('STBHT =',E14.6)
0612 C
0613 C DRAW AN ELLIPSE FOR THE EFFECTIVE CLOUD HEIGHT
0614 C ---HCEN AND KCEN ARE THE COORDINATES OF THE CENTER OF THE ELLIPSE
0615 C --AAX AND BAX ARE THE MAJOR AND MINOR AXES
0616 C -- THE SIGN OF THE X VALUE IS CHANGED EVERY OTHER TIME THE LOOP
0617 C IS EXECUTED
0618 C -- THE Y VALUE IS DECREMENTED EVERY OTHER TIME THE LOOP IS EXECUTED
0619 C
0620 HCEN = 400.0
0621 KCEN = STBHT * 0.00 + 90.0
0622 AAX = <GAMMAX+STBALT*0.00>*0.0
0623 BAX = <0.5 * <GAMMAX+STBALT - STBALT + ALT<LAYTOP>>>*0.00
0624 YINC = BAX/25.0
0625 SIGN = 1.0
0626 YPT = KCEN + BAX
0627 XPT = HCEN
0628 YELL(1) = YPT
0629 XELL(1) = HCEN
0630 JDEC = 1
0631 JSW = 1
0632 DO 99 K=2,100
0633 IF<JDEC.EQ.1> YPT = YPT - YINC
0634 RAD = AAX * SORT<1.0 - <YPT-KCEN>**2>/<BAX**2>>
0635 XELL(K) = HCEN + SIGN * RAD
0636 YELL(K) = YPT
0637 JSW = JSW + 1
0638 IF<JSW.EQ.2> SIGN = -SIGN
0639 IF<JSW.EQ.2> JSW = JSW - 2
0640 99 JDEC = -JDEC
0641 CALL LINE<XELL,YELL,100,0>
0642 C
0643 C** SPECIFY THE DATA FILE USED.
0644 C
0645 CALL CHAR<64.0,40.0,0.0,FILE,6,2,1>
0646 C
0647 C IF REQUESTED, WRITE OUT THE BOTTOM OF THE SURFACE LAYER
0648 C LINE AND ALLOW IT TO BE MOVED UP AND DOWN
0649 C
0650 IF<CALC.ME.1>GO TO 100
0651 CALL MOVEN<LAYBOT,ILP,1,IBOT,414.0,BSHFX,5>
0652 BOTLAY = ALT<LAYBOT>
0653 CALHT = 89TLAY
0654 C
0655 C IF NOT RUNNING ON THE PLASMASCOPE, DO NOT ASK IMMEDIATELY QUESTION
0656 C
0657 100 IF<IVCRT>GO TO 110
0658 .CH 24. 0.1 (1). 1.0)

```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0659 CALL CHARC(160,0,16,0,0, IHARD(10),0,0,0)
0660 CALL CHARC(32,0,16,0,0, IHARD(14),6,0,0)
0661 CALL IJK(1, JTYPE,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0)
0662 IF(IJK .GT. 15)GO TO 110
0663 CALL RWDIS(NAME,1)
0664 CALL NGRAF
0665 CALL EXEC(9,AMETQ)
0666 CALL RWDIS(NAME,0)
0667 C
0668 C WRITE OUT THE COMMON DISK FILE
0669 C
0670 C 110 CALL RWDIS(NAME,1)
0671 C
0672 C CALL CLEAR AND NGRAF TO CLEAR PLASMASCOPE AND TERMINATE
0673 C GRAPHIC MODE
0674 C
0675 CC CALL CLEAR
0676 IF(LVRPLS) CALL CLEAR
0677 CALL NGRAF
0678 C
0679 C
0680 C*** TERMINATE RMTPS ***
0681 $TOP
0682 END
0683 END*
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```

0001 T=00003 IS ON CR00003 USING 00052 BLKS R=0000
0002 SUBROUTINE RUDTIS(NAME, JJ), *REEDS, CAB810015
0003
0004 -----
0005
0006 THIS SUBROUTINE READS AND WRITES COMMON AREAS TO DISC
0007 TO BE UTILIZED BY THE VARIOUS REED PROGRAMS.
0008 -----
0009
0010 >>> COMMON AREAS
0011
0012 >>> MATH PARAMETERS
0013 COMMON PI, F10VK2, F143, TU0P1, SOR2PI, DEGRAD, RADDEG, IV2
0014
0015 >>> VEHICLE PARAMETERS
0016 COMMON VPAR<10>, SIGHCL
0017
0018 >>> OPTION PARAMETERS
0019 COMMON YPOS, YE, ISNDFD, IWHICL, IRUN, ICAC, IPLACE, ITIMEZ, IGOIMP,
0020 NUMRUN, LU, NUM, MONLY, MFORM, LUPAR<1>, ITDU, IPL1, IPL2, IPL3
0021
0022 >>> MET DATA
0023 COMMON ALT<30>, IDIR<30>, SPEED<30>, TEMP<30>, PRESS<30>, PTEMP<30>,
0024 SURDEN, FILE<3>
0025
0026 >>> CALCULATION TIME
0027 COMMON ITIME, IDAY, IMONTH<2>, IYEAR
0028
0029 >>> SOUNDING TIME
0030 COMMON ISTIME, ISDAY, ISMONK<2>, ISYEAR
0031
0032 >>> LAUNCH TIME
0033 COMMON LTIME, LTIM, LDAY, LMONTH<2>, LYEAR, LAUNTD<10>
0034
0035 >>> CLOUD STABILIZATION AND LAYER PARAMETERS
0036 COMMON STBAL, STBHT, STBZD, STBRNG, STBTM, CLDRAD, RISTMK<30>, BOTLAY,
0037 TOPLAY, CALHT, LAYTOP, LAYBOT, REFLEC, IPTZ
0038
0039 >>> DIFFUSION COEFFICIENTS
0040 COMMON SIX0, SIXX, SIGZ, SIG20, SIGAP, XDIST, YDIST, EXPZ, SUSIGZ, SIGY, SIGAL,
0041 CLDLG, AVSFD, AYDIR, SIGAZ
0042
0043 >>> CONCENTRATION AND DOSAGE VALUES AND RANGES
0044 COMMON CONMAX, CONCEK, DOSMAX, DOSPK, RINGSMC, RINGSND
0045
0046 >>> EQUIVALENCE STATEMENT FOR VEHICLE PARAMETERS
0047 EQUIVALENCE (QC1, VPAR<1>), (QC2, VPAR<2>), (QC3, VPAR<3>),
0048 (QT1, VPAR<4>), (QT2, VPAR<5>), (QT3, VPAR<6>),
0049 (AA, VPAR<7>), (BB, VPAR<8>), (CC, VPAR<9>),
0050 (HEATH, VPAR<10>), (HEATH, VPAR<11>), (HEATA, VPAR<12>),
0051 (PNCL, VPAR<13>), (PC0, VPAR<14>), (PC02, VPAR<15>),
0052 (F1, VPAR<16>), (F2, VPAR<17>), (HHW, VPAR<18>),
0053
0054
0055
0056
0057
0058
0059
0060
0061
0062
0063
0064
0065
0066
0067
0068
0069
0070
0071
0072
0073
0074
0075
0076
0077
0078
0079
0080
0081
0082
0083
0084
0085
0086
0087
0088
0089
0090
0091
0092
0093
0094
0095
0096
0097
0098
0099
0100

```



```

0119 COMMON LTIME, LTIM, LDAY, LMONTH(2), LYEAR, LAUHTD(10)
0120 LOGICAL LTIME, LVERSH
0121 C
0122 C >>> CLOUD STABILIZATION AND LAYER PARAMETERS
0123 COMMON STBLT, STBT, STBZD, STBRG, STBTIN, CLDRAD, RSTIK(30), BOTLAY,
0124 TOPLAY, CALHT, LAYTOP, LAYBOT, REFLEC, IPTZ
0125 C
0126 C >>> DIFFUSION COEFFICIENTS
0127 COMMON SIGX0, SIGX, SIGZ0, SIGAP, XDIST, YDIST, EXPZ, SOSIGZ, SIGY, SIGAL,
0128 CLDLG, AVSPD, AVDIR, SIGAZ
0129 C
0130 C >>> CONCENTRATION AND DOSAGE VALUES AND RANGES
0131 COMMON CONMAX, CONCPK, DOSMAX, DOSPK, RINGSZ, RINGSZD
0132 C
0133 C >>> EQUIVALENCE STATEMENT FOR VEHICLE PARAMETERS
0134 EQUIVALENCE (QC1, VPAR(1)), (QC2, VPAR(2)), (QC3, VPAR(3)),
0135 (QT1, VPAR(4)), (QT2, VPAR(5)), (QT3, VPAR(6)),
0136 (HA, VPAR(7)), (BB, VPAR(8)), (CC, VPAR(9)),
0137 (HEATN, VPAR(10)), (HEATM, VPAR(11)), (HEATA, VPAR(12)),
0138 (PHCL, VPAR(13)), (PCD, VPAR(14)), (PCO2, VPAR(15)),
0139 (PAL203, VPAR(16)), (PH0, VPAR(17)), (GAMMAX, VPAR(18))
0140 C
0141 C*** FORMATS
0142 91 FORMAT(A2)
0143 99 FORMAT(2X, " UP/DOWN/CONT. SECTION ***IANS= ", A2, " IX, IY=", 2I3)
0144 2000 FORMAT (F6.1)
0145 2001 FORMAT (" = ", 13" .0")
0146 INTEGER QUES(13), ANS1, ANS2(2), ANS3(4), BLANKS(26)
0147 DIMENSION LAB(1), XLINE(1), YLINE(2), JNDALT(3), JNDVAR(3,4)
0148 EQUIVALENCE (JNDV1, JNDVAR(1,1)), (JNDV2, JNDVAR(1,2)),
0149 (JNDV3, JNDVAR(1,3)), (JNDV4, JNDVAR(1,4))
0150 DATA QUES/2HMO, 2HVE, 2H , 2H , 2H 0.2HF , 2HSU, 2HNF, 2HAC, 2HE , 2HLA,
0151 2HYE, 2HIR, /
0152 DATA ANS1/2HUP/, ANS2/2HDO, 2HUN/, ANS3/2HCO, 2HNT, 2HII, 2HUE,
0153 DATA BLANKS/26*2H /
0154 C
0155 C*** SET LU FLAG
0156 CALL VERSK(I)
0157 LVERSH = I .EQ. 7
0158 C
0159 C*** INITIALIZE NEW ALTITUDE LEVEL INDEX
0160 NEWJND = 0
0161 C
0162 C*** START LOOP TO FIND DESIRED LEVEL
0163 1 YLINE(1) = ALT(JND) * 0.08 + 90.0
0164 YLINE(2) = YLINE(1)
0165 C
0166 C*** DRAW LAYER
0167 DO 4 I=1, NLINE
0168 J = 2 + I - 1
0169 4 CALL LINE(XLINE(I), YLINE(2, 0))
0170 C
0171 C*** IF FIRST PASS, WRITE QUESTION... OTHERWISE SKIP TO ANSWER
0172 IF(NEWJND .EQ. JND) GO TO 11
0173 QUES(3) = LAB(1)
0174 QUES(4) = LAB(2)
0175 CC
0176 CC*** ATTEMPT TO PREVENT OVERWRITING ON CRT VERSION... JSH
0177
0178

```



ORIGINAL PAGE IS  
OF POOR QUALITY

```
0179 IF(LVERSH) Y = 1.0
0180 CC CALL CHARC(0.0,1.0,-1,GUES,26,3,0)
0181 CALL CHARC(0.0,Y,-1,GUES,26,3,0)
0182 CC CALL CHARC(29.0,1.0,-1,ANS1,2,0,0)
0183 CALL CHARC(29.0,Y,-1,ANS1,2,0,0)
0184 CC CALL CHARC(36.0,1.0,-1,ANS2,4,0,0)
0185 CALL CHARC(36.0,Y,-1,ANS2,4,0,0)
0186 CC CALL CHARC(43.0,1.0,-1,ANS3,6,3,0)
0187 CALL CHARC(43.0,Y,-1,ANS3,6,3,0)
0188 II CONTINUE
0189 C
0190 C*** THIS IS THE UP/DOWN/CONTINUE DECISION AREA
0191 C
0192 C*** IF PLASMASCOPE, GO TO TOUCH INPUT
0193 IF(LVERSH) GO TO 12
0194 C
0195 C*** DEFAULT IS UP
0196 IX = 10
0197 READ(LU,91)IANS
0198 IF( IANS.NE.2HDD) .AND. ( IANS.NE.2HCO) ) GO TO 13
0199 C
0200 C*** THIS IS CONTINUE OR DOWN
0201 IX = 25
0202 IF( IANS.NE.2HDD ) GO TO 13
0203 C
0204 C*** THIS IS DOWN
0205 IX = 16
0206 GO TO 13
0207 II CONTINUE
0208 CALL IN(1,J,0.0,0.0,0.0,0.31,0.31,IX,IY)
0209 II CONTINUE
0210 C
0211 C*** IF NOT CONTINUE, MOVE THE LINE--BUT DO NOT LABEL IT OR FILL IN THE TABLE
0212 WRITE(6,99) IANS,IX,IY
0213 IF(IX .LE. 20)GO TO 25
0214 C
0215 C*** IF CONTINUE, LABEL THE LINE AND FILL IN THE TABLE
0216 Y = YLINE(1) + 2.0
0217 CALL CHAR(460.0,Y,0,LAB,4,2,1)
0218 Y = Y - 10.0
0219 CALL CODE
0220 WRITE ( JHDN,T,2000) ALT(JHD)
0221 CALL CHAR(460.0,Y,0,JNDALT,6,2,1)
0222 CALL CODE
0223 WRITE ( JHDV1,2000) TEMP(JHD)
0224 YLABEL = PTEMP(JHD) - 273.16
0225 CALL CODE
0226 WRITE ( JHDV2,2000) YLABEL
0227 CALL CODE
0228 WRITE ( JHDV3,2000) SPEED(JHD)
0229 CALL CODE
0230 WRITE ( JHDV4,2001) IDIR(JHD)
0231 YLABEL = 450.0
0232 DO 14 I=1,4
0233 CALL CHAR(XLABEL,YLABEL,0,JHDV4(I),6,2,1)
0234 YLABEL = YLABEL - 10.0
0235 C 22 IF(NEHJHD .EQ. JHD)GO TO 11
0236 C GO 24 I=1,NLINE
0237 C J = 2 * I - 1
0238 C 24 CALL LINE(XLINE(J),YLINE,2,1)
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0239 CALL CHAR(460.0,Y,0,JNDALT,6,1,1)
0240 Y = Y + 10.0
0241 CALL CHAR(460.0,Y,0,LAB,4,1,1)
0242 YLABEL = 450.0
0243 DO 26 I=1,4
0244 CALL CHAR(XLABEL,YLABEL,0,JNDVAR(1,1),6,1,1)
0245 YLABEL = YLABEL - 10.0
0246 CALL CHAR(0.0,1.0,-1,BLANKS,51,0.0)
0247 RETURN
0248
0249 IF(IX .GE. 17)GO TO 29
0250 NEWJND = MIN(JND + 1,MAXJND)
0251 JND = NEWJND
0252 GO TO 1
0253
0254 NEWJND = MAX(JND - 1,MINJND)
0255 JND = NEWJND
0256 GO TO 1
0257 END
0258 SUBROUTINE WINDS(WD,NWD,ISC)
0259 C
0260 C
0261 C -----
0262 C - THIS SUBROUTINE COMPUTES THE WIND DIRECTION LABELS -
0263 C - FOR PLOTTING -
0264 C -----
0265 DIMENSION WD(1),ENDPT(4),NUMUP(4)
0266 EQUIVALENCE (J,LEAST)
0267 DATA ENDPT/0.0,90.0,180.0,270.0/
0268 DO 2 I=1,4
0269 NUMUP(I) = 0
0270 WD2 = WD(I)
0271 DO 8 I=2,NWD
0272 WD1 = WD2
0273 WD2 = WD(I)
0274 DO 6 J=1,4
0275 C1 = WD1 - ENDPT(J)
0276 IF(C1 .LT. 0.0)C1 = C1 + 360.0
0277 C2 = WD2 - ENDPT(J)
0278 IF(C2 .LT. 0.0)C2 = C2 + 360.0
0279 IF(ABS(C1-C2) .LE. 180.0)GO TO 6
0280 NUMUP(J) = NUMUP(J) + 1
0281
0282 6 CONTINUE
0283 8 CONTINUE
0284 ISC = 1
0285 LEAST = NUMUP(1)
0286 DO 12 I=2,4
0287 IF(NUMUP(I) .GE. LEAST)GO TO 12
0288 ISC = I
0289 LEAST = NUMUP(I)
0290 12 CONTINUE
0291 DO 17 I=1,NWD
0292 WD(I) = WD(I) - ENDPT(ISC)
0293 IF(WD(I) .LT. 0.0)WD(I) = WD(I) + 360.0
0294 17 CONTINUE
0295 WD2 = WD(I)
0296 DO 22 I=2,NWD
0297 WD1 = WD2
0298 WD2 = WD(I)
0299 22 CONTINUE
0300 IRS( WD1, WD2)
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```

0299      UD(1) = UD(1)
0300      22 CONTINUE
0301      RETURN
0302      END
0303      SUBROUTINE CLOUD(XP,YP)
0304      C
0305      C
0306      C
0307      C --- THIS SUBROUTINE DRAWS A CIRCULAR CLOUD FOR THE MET
0308      C --- PROFILE, AT A SPECIFIED X AND Y POSITION.
0309      C
0310      C -----
0311      C >>> COMMON AREAS
0312      C
0313      C
0314      C >>> MATH PARAMETERS
0315      C
0316      C COMMON P1,PIOR2,F143,IUOF1,SOR2PI,DEGRAD,PARDEG,IV2
0317      C
0318      C >>> VEHICLE PARAMETERS
0319      C COMMON VPAR(18),SIGLCL
0320      C
0321      C >>> OPTION PARAMETERS
0322      C COMMON YPOS,YES,ISMOFO,IVNICL,IRUN,ICALC,IPLACE,ITIMEZ,IGOTHP,
0323      C NUMRUN,LU,NUM,MOHLY,MFORM,LUPKHT,ITDU,IPL1,IPL2,IPL3
0324      C INTEGER YES
0325      C
0326      C >>> MET DATA
0327      C COMMON ALT(30),IDIR(30),SPEED(30),TEMP(30),PRESS(30),PTEMP(30),
0328      C SURDEH,FILE(3)
0329      C INTEGER FILE
0330      C
0331      C >>> CALCULATION TIME
0332      C COMMON ITIME,IDAY,IMONTH(2),IYEAR
0333      C
0334      C >>> SOUNDING TIME
0335      C COMMON ISTIME,ISDAY,ISMONK(2),ISYEAR
0336      C
0337      C >>> LAUNCH TIME
0338      C COMMON LTIME,LTIM,LDAY,LMONTH(2),LYEAR,LAUNTD(10)
0339      C LOGICAL LTIME
0340      C
0341      C >>> CLOUD STABILIZATION AND LAYER PARAMETERS
0342      C COMMON STBAL,STBHT,STBAZD,STBRHG,STBTIM,CLODRAD,RISYIM(30),BOTLAY,
0343      C TOPLAY,CALHT,LAYTOP,LAYBOT,REFLEC,IPIZ
0344      C
0345      C >>> DIFFUSION COEFFICIENTS
0346      C COMMON SIGX,SIGY,SIGZ0,SIGAP,XDIST,YDIST,EXPZ,SOSIGZ,SIGY,SIGAL,
0347      C CLOLNG,AVSPD,AVDIR,SIGAZ
0348      C
0349      C >>> CONCENTRATION AND USAGE VALUES AND RANGES
0350      C COMMON CONMAX,CONCPK,DOSHAX,DOSPK,RIGSHC,RIGSHD
0351      C
0352      C >>> EQUIVALENCE STATEMENT FOR VEHICLE PARAMETERS
0353      C EQUIVALENCE (OC1,VPAR(1)),(OC2,VPAR(2)),(OC3,VPAR(3)),
0354      C (QT1,VPAR(4)),(QT2,VPAR(5)),(QT3,VPAR(6)),
0355      C (AA,VPAR(7)),(BB,VPAR(8)),(CC,VPAR(9)),
0356      C (HEATN,VPAR(10)),(HEATM,VPAR(11)),(HEATA,VPAR(12)),
0357      C (PHCL,VPAR(13)),(P10,VPAR(14)),(P11,VPAR(15)),
0358      C (PAL,203,VPAR(16)),(P16,VPAR(17)),(LANNHAX,VPAR(18))

```

```

0359 DIMENSION X(61),Y(61)
0360 C
0361 C THE MULTIPLICATIVE CONSTANT .0 WAS INTRODUCED TO DRAW A CIRCULAR
0362 C CLOUD WITH XSCALE IN USE.
0363 C
0364 RADIUS = GAMMAX * STBALT * 0.08
0365 DO 7 I=1,61
0366 X(I) = (RADIUS * COS(.01745329252 * FLOAT(6 * I))) * .0 + XP
0367 Y(I) = RADIUS * SIN(.01745329252 * FLOAT(6 * I)) + YP
0368 CALL LINE(X,Y,61,0)
0369 RADIUS = 5.0
0370 X(1) = XP + RADIUS
0371 X(2) = XP
0372 X(3) = XP - RADIUS
0373 X(4) = XP
0374 X(5) = X(1)
0375 Y(1) = YP
0376 Y(2) = YP + RADIUS
0377 Y(3) = YP
0378 Y(4) = YP - RADIUS
0379 Y(5) = Y(1)
0380 CALL LINE(X,Y,5,0)
0381 X(2) = XP - RADIUS
0382 Y(2) = YP
0383 CALL LINE(X,Y,2,0)
0384 X(3) = XP
0385 Y(3) = YP + RADIUS
0386 CALL LINE(X,Y,3,2,0)
0387 RETURN
0388 END
0389 END$

```

ORIGINAL PAGE IS  
OF POOR QUALITY

ORIGINAL PAGE IS  
OF POOR QUALITY

ARCINS T-9003 IS GH CR0003 USING 00073 DLMS R=0000

```
0001 FTMX,L
0002 PROGRAM RCONS(3),REDS CONC. & DOSAGE PROG...UPDATED 12 22 79, CBJ
0003 C
0004 C
0005 C
0006 C * CONCENTRATION AND DOSAGE PROGRAM -- A PROGRAM OF THE
0007 C * REEDS SERIES OF PROGRAMS
0008 C *
0009 C
0010 C
0011 C >>> COMMON AREAS
0012 C
0013 C
0014 C
0015 C >>> MATH PARAMETERS
0016 C COMMON PI,PIOVK2,P143,TMOPI,SQR2P1,DEGRAD,RADEG,IV2
0017 C
0018 C >>> VEHICLE PARAMETERS
0019 C COMMON VPAR(18),SICHEL
0020 C
0021 C >>> OPTION PARAMETERS
0022 C COMMON YPOS,YES,ISNOFG,IVHICL,IRUN,ICALC,IFLACE,ITIMEZ,IGOTMP,
0023 C MURRUN,LU,NUM,MONLY,MFORM,LUPRNT,ITDU,IPL1,IPL2,IPL3
0024 C INTEGER YES
0025 C
0026 C >>> MET DATA
0027 C COMMON ALI(30),IDIR(30),SPEED(30),TEMP(30),PRESS(30),PTEMP(30),
0028 C SURDEN,FILE(3)
0029 C
0030 C
0031 C >>> CALCULATION TIME
0032 C COMMON ITIME,IDAY,IMONTH(2),IYEAR
0033 C
0034 C >>> SOUNDING TIME
0035 C COMMON ISTIME,ISDAY,ISMON(2),ISYEAR
0036 C
0037 C >>> LAUNCH TIME
0038 C COMMON LTIME,LTIM,LDAY,LMONTH(2),LYEAR,LAUNTD(10)
0039 C LOGICAL LTIME, LVERSH
0040 C
0041 C >>> CLOUD STABILIZATION AND LAYER PARAMETERS
0042 C COMMON STBLT,STBLT,STBZD,STBRG,STBTM,CLOPAD,RISTIM(30),BOTLAY,
0043 C TOPLAY,CALHT,LAYTOP,LAYBOT,REFLEC,IPTZ
0044 C
0045 C >>> DIFFUSION COEFFICIENTS
0046 C COMMON SIGX,SIGZ,SIGZ0,SIGAP,XDIST,YDIST,EXPZ,SUSIGZ,SIGY,SIGAL,
0047 C CDLNG,AVSPD,AVDIR,SIGAZ
0048 C
0049 C >>> CONCENTRATION AND DOSAGE VALUES AND RANGES
0050 C COMMON CONMAX,CONCPK,DOSEMAX,DOSEK,PIGSMC,PIGSHD
0051 C
0052 C >>> EQUIVALENCE STATEMENT FOR VEHICLE PARAMETERS
0053 C EQUIVALENCE (QC1,VPAR(1)),(QC2,VPAR(2)),(QC3,VPAR(3)),
0054 C (QT1,VPAR(4)),(QT2,VPAR(5)),(QT3,VPAR(6)),
0055 C (QA,VPAR(7)),(QB,VPAR(8)),(CC,VPAR(9)),
0056 C (HEATH,VPAR(10)),(FEATH,VPAR(11)),(CREATA,VPAR(12)),
0057 C (PICL,VPAR(13)),(PCO,VPAR(14)),(PCO,VPAR(15)),
0058 C (PAL203,VPAR(16)),(PMO,VPAR(17)),(PAH100,VPAR(18))
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0059 C          OUTPUT FORMAT STATEMENTS
0060 C
0061 C
0062 90      FORMAT(A1)
0063      200 FORMAT ("1-12X" CLOUD CONCENTRATIONS AND DOSAGES"/
0064      "0DISTANCE"4X"CONCENTRATION"5X"DOSAGE"6X
0065      "TIME AFTER LAUNCH(SEC)",11X,"POSITION FROM PAD"/
0066      " (METERS)"8X"(PPM)"9X"(PPM SEC)"6X"START"3X"FINISH",
0067      " RANGE
0068      201 FORMAT (1X"1.8XF7.3,8XF7.3,9XF5.1,3XF5.1,16X,2F8.1)
0069      202 FORMAT (/"0"0"0"POINT OF MAXIMUM CONCENTRATION"0000"/
0070      "6X"RANGE FROM PAD(M); "F8.1/"
0071      "6X"DIRECTION(DEG); "F5.1/"
0072      "6X"HEIGHT(M); "F5.1/"
0073      "6X"MAXIMUM CONCENTRATION(PPM); "F6.3)
0074      203 FORMAT (/"0"0"0"POINT OF MAXIMUM DOSAGE"0000"/
0075      "6X"RANGE FROM PAD(M); "F8.1/"
0076      "6X"DIRECTION(DEG); "F5.1/"
0077      "6X"HEIGHT(M); "F6.1/"
0078      "6X"MAXIMUM DOSAGE (PPM SEC); "F6.3)
0079      204 FORMAT (/"0"0"0"CONCENTRATIONS AND DOSAGES WITH 10 DEGREE "
0080      "UNCERTAINTIES"000"/
0081      205 FORMAT ("0"5X"RANGE(M); "F7.1/"
0082      "6X"AZIMUTH(DEG); "F5.1/"
0083      "6X"MATRIAL"5X"CONCENTRATION(PPM)"11X"DOSAGE(PPM)"0"/
0084      206 FORMAT (415.12)
0085 C 207 FORMAT (7X3A2,6XF8.3" +/- "F8.3,4XF8.3" +/- "F8.3)
0086 C 207 FORMAT (7X3A2,6XE8.3" +/- "F8.3,4XF8.3" +/- "F8.3)
0087 C
0088 C
0089 C
0090 C
0091 C          INTEGER RISOP(3)
0092 C          DIMENSION FACT(3),CHMPL(3),DMHPL(3),MATS(3,5),NAME(3),
0093 C          NAMEF(3),ILINE(32),IDATAF(10),IERS(32),
0094 C          XD19TV(81),DOSV(81),CONCV(61)
0095 C          DATA STATEMENTS
0096 C          DATA NAME,2H=R,2HEE,2HD2/,NAMEF/2H?R,2HCO,2HNS/
0097 C          DATA IERS/32*2H /
0098 C
0099 C          DATA FACT/0.0,-0.174533,0.174533/
0100 C          DATA MATS/2H ,2HHC,2HL ,2H ,2H C,2HO ,
0101 C          2H ,2HCO,2H2 ,2H A,2HL2,2H03,
0102 C          2H ,2H N,2HO /
0103 C          DATA RISOP/2H1,2H50,1HS/
0104 C
0105 C          CALL GRAF TO INITIALIZE SCOPE (APPROPRIATE ONLY WHEN USING
0106 C          PLASMASCOPE)
0107 C
0108 C          CALL GRAF(1)
0109 C
0110 C          FIND OUT IF THIS IS THE CRT OR PLASMASCOPE VERSION OF REED2
0111 C          BEING RUN
0112 C
0113 C          CALL VERSH(1)
0114 C          LVERSN = 1 .EQ. 0
0115 C
0116 C          READ COMMON DISK FILE =REED
0117 C
0118 C          CALL RUDIS(NAME,0)
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0119 C DO LOOP FOR CONCENTRATION AND DOSAGE CALCULATIONS
0120 C
0121 C XDIST - RANGE FROM STABILIZATION
0122 C D0SPK - DOSAGE
0123 C D0SMAX - MAXIMUM DOSAGE
0124 C CONCPK - CONCENTRATION
0125 C CONMAX - MAXIMUM CONCENTRATION
0126 C
0127 C
0128 C NUMV = 0
0129 C YES = 1HY
0130 C MO = 1MN
0131 C CONMAX = 0.0
0132 C D0SMAX = 0.0
0133 C ACTVOL = P143 * CLDRAD * CLDRAD * CLDRAD
0134 C TOTVOL = ACTVOL
0135 C IF(I<IV2 .EQ. 1)ACTVOL = PI * (ALT(LAYTOP) * CLDRAD - STBALT)**2 *
0136 C (2.0 * CLDRAD - ALT(LAYTOP) + STBALT)/3.0
0137 C
0138 C SIGNCL = SIGNCL * ACTVOL/TOTVOL
0139 C
0140 C *** PREPARE CONSTANTS FOR RELATIVE PAD/CENTERLINE CALCULATIONS
0141 C STBAZR = DEGRAD * STBAZD
0142 C COSSTB = COS(PI - STBAZR)
0143 C SINSTB = SIN(PI - STBAZR)
0144 C YSTAB = STBRNG * COSSTB
0145 C AVDIRR = DEGRAD * AVDIR
0146 C COSAV = COS(PI - AVDIR)
0147 C SINAV = SIN(PI - AVDIR)
0148 C WRITE (LUPRINT,200)
0149 C
0150 C *** SET OFF-CENTERLINE DISTANCE TO 0 FOR CENTERLINE CALCULATION
0151 C YDIST = 0.0
0152 C DO 59 I=0,20000.250
0153 C
0154 C INUV = NUMV + I
0155 C XDIST = I
0156 C
0157 C *** GET PAD-RELATIVE RANGE AND AZIMUTH TO APPROPRIATE CENTERLINE POSITION
0158 C XREL = XSTAB + XDIST * COSAV
0159 C YREL = YSTAB + XDIST * SINAV
0160 C SITRNG = SORT(XREL * XREL + YREL * YREL)
0161 C DIRPCL = ATAN2(YREL, XREL)
0162 C SITAZD = RADDEC * (PI - DIRPCL)
0163 C XDISTV(NUMV) = XDIST
0164 C
0165 C CALL DFEXP(LAYTOP, 1000.0)
0166 C
0167 C THE WRITE STATEMENTS BELOW HAVE BEEN ADDED AS TEMPORARY
0168 C AIDS IN VALIDATING CALCULATIONS FOR DOSAGE AND CONCENTRATION.
0169 C
0170 C D0SPK = SIGNCL * EXPZ/(MOPI * SIGY * AVSPD * SORT(0.5 * SOSIGZ))
0171 C D0SV(NUMV) = D0SPK
0172 C CONCPK = D0SPK * AV3PD/(SOR2PI * SIGX)
0173 C CONCV(NUMV) = CONCPK
0174 C
0175 C IF(D0SPK .LE. D0SMAX)GO TO 55
0176 C RMESHM = XDIST
0177 C D0SMAX = D0SPK
0178 C
```





ORIGINAL FROM IS  
OF POOR QUALITY

```
0239 IEXPC = EX
0240 IF(EN .LT. 0.0)IEXPC = IEXPC - 1
0241 IEXPC = - IEXPC
0242 CALL GPLOT(XDISTY,DOSV,CONCV,HUMY,IEXPD,IEXPC)
0243 CALL LABEL(IEXPD,IEXPC)
0244 C
0245 C CALCULATE AND WRITE OUT THE POINT OF MAXIMUM CONCENTRATION
0246 C
0247 61 CONTINUE
0248 XDIST = RIGSMC * COSAV + XSTAB
0249 YDIST = RIGSMC * SINAV + YSTAR
0250 RIGPMC = SORT(XDIST * XDIST + YDIST * YDIST)
0251 ARG1 = ATAN2(YDIST,XDIST)
0252 DIRPMC = PI - ARG1
0253 DIRPMC = RADDEG * DIRPMC
0254 WRITE (LUPRINT,202) RIGPMC,DIRPMC,CALHT,CONMAX
0255 C
0256 C CALCULATE AND WRITE OUT THE POINT OF MAXIMUM DENSITY
0257 C
0258 XDIST = RIGSMO * COSAV + STARGC * COSSTB
0259 YDIST = RIGSMO * SINAV + STARGC * SINSTB
0260 RIGPMO = SORT(XDIST * XDIST + YDIST * YDIST)
0261 ARG1 = ATAN2(YDIST,XDIST)
0262 DIRPMO = PI - ARG1
0263 DIRPMO = RADDEG * DIRPMO
0264 WRITE (LUPRINT,203) RIGPMO,DIRPMO,CALHT,DOSHAX
0265 C
0266 C IF THIS IS A PRODUCTION RUN, SKIP THE OFF CENTER CONCENTRATION
0267 C SECTION AND THE CALL OF PROGRAM RISOP -- IF NOT, DO OFF CENTER
0268 C CONCENTRATIONS IF DESIRED
0269 C
0270 C IF(IRUN .EQ. 3)GO TO 80
0271 C
0272 C ARE OFF CENTER CONCENTRATIONS DESIRED?
0273 C
0274 CALL DREAD(NAMEF,3,ILINE)
0275 CALL LERS(YPOS,LVERSH)
0276 CALL CHAR(0,,YPOS,0,ILINE,30,3,0)
0277 CALL CHAR(384,0,YPOS,0,ILINE(25),8,3,0)
0278 CALL CHAR(464,0,YPOS,0,ILINE(30),6,0,0)
0279 C
0280 C IF PLASMASCOPE VERSION, SKIP TO THE CALL TO "IN".
0281 C
0282 C IF(LVERSH) GO TO 612
0283 C
0284 C OTHERWISE, RESPOND WITH Y FOR YES, N FOR NO TO ANSWER
0285 C QUESTIONS.
0286 C
0287 READ(LU,90) IANS
0288 IF(IANS.EQ.100)STOP
0289 C
0290 C "YES" IS THE DEFAULT CASE
0291 C
0292 C IX = 25
0293 C
0294 C THIS IS THE "NO" RESPONSE
0295 C
0296 IF(IANS .NE. NO) GO TO 613
0297 IX = 30
0298 GO TO 613
```



ORIGINAL PAGE IS  
OF POOR QUALITY

```
0359 YPOS = YPOS - 16.  
0360 IF(LVERSH) YPOS = YPOS - 16.  
0361 IF(YPOS .LT. 48.0.AND.LVERSH) YPOS = 458  
0362 IF(LVERSH) CALL LERS(YPOS,LVERSH)  
0363 CALL CHAR(0.0,YPOS,0,ILINE,64,0.0)  
0364 IF(LVERSH)GO TO 356  
0365 JXAX = 14  
0366 IPOS = -2  
0367 WRITE(LU,351)JXAX,IPOS  
0368 FORMAT(" 12c"13"R")  
0369 MIN = 7  
0370 CALL BLANK(IDATAF,10)  
0371 CALL INK(0,JTYPE,112.,YPOS,0,IDATAF,MIN,0.31,0.31,IX,IY)  
0372 CALL CODE  
0373 READ (IDATAF,*) SITRNG  
0374 IF(SITRNG .LE. 0.0)GO TO 78  
0375 IF(LVERSH)GO TO 365  
0376 JXAX = 34  
0377 WRITE(LU,360)JXAX,IPOS  
0378 FORMAT(" 12c"13"R")  
0379 MIN = 7  
0380 CALL BLANK(IDATAF,10)  
0381 CALL INK(0,JTYPE,272.,YPOS,0,IDATAF,MIN,0.31,0.31,IX,IY)  
0382 CALL CODE  
0383 READ (IDATAF,*) SITAZD  
0384 IF(YPOS.LT.0.0)GO TO 380  
0385 YPOS = YPOS - 16.  
0386 IF(YPOS .LT. 48.0)YPOS = 458.0  
0387 IF(YPOS .LT. 48.0.AND.LVERSH)YPOS = 458.0  
0388 J80 WRITE (LUPRINT,205) SITRNG,SITAZD  
0389 C  
0390 .73 SITAZR = DEGRAD * SITAZD  
0391 COSSIT = COS(PI - SITAZR)  
0392 SINSIT = SIN(PI - SITAZR)  
0393 XSITEP = SITRNG * COESIT  
0394 YSITEP = SITRNG * SINSIT  
0395 C  
0396 C ON THE PLOTTER, WRITE OUT AN ASTERISK AND THE ITERATION  
0397 C NUMBER AT THE LOCATION WHERE THE OFF CENTER CONCENTRATION  
0398 C CALCULATION IS DESIRED  
0399 C  
0400 IXPLOT = IXSET + 0.2631 * XSITEP  
0401 IYPLOT = IYSET + 0.3545 * YSITEP  
0402 WRITE (IPL3) -1,1,IXPLOT,IYPLOT  
0403 CALL SYMBL(100,IPL25,1H*,IPL3)  
0404 IXPLOT = IXPLOT + 75  
0405 WRITE (IPL3) -1,1,IXPLOT,IYPLOT  
0406 WRITE (IPL3,206) 100,0,0.125,ITER  
0407 C  
0408 C CALCULATE THE CONCENTRATIONS AND DOSAGES AT THIS POINT PLUS  
0409 C 10 DEGREES UNCERTAINTIES ON EITHER SIDE  
0410 C  
0411 C*** CALCULATE DISTANCE/ANGLE FROM STABILIZATION TO SITE OF INTEREST  
0412 XDEL = XSITEP - XSTAB  
0413 YDEL = YSITEP - YSTAB  
0414 RDEL = SQRT(XDEL**2 + YDEL**2)  
0415 BETAF = PI - ATN2(YDEL,XDEL)  
0416 C416 CHECK = ATAN2(YDEL,XDEL)  
0417 BET = RADNEG * BETAF  
0418 C
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0419 DO 74 I=1,3
0420 ARG1 = BETAF - FACT(I)
0421 C
0422 C*** CALCULATE DISTANCE FROM CLOUD CENTER TO SITE OF INTEREST
0423 YCCSIT = RDEL * SIN(ARG1)
0424 XCCSIT = RDEL * COS(ARG1)
0425 XDIST = XCCSIT
0426 CALL DFEXP(LAYTOP,1000.0)
0427 C
0428 C*** EXP(-Y * Y) IS NOT DEFINED...YCCSIT IS BEING SUB'D FOR Y
0429 C*** Y IS PRINTED JUST TO PROVE THE CASE
0430 WRITE(6,997)BET,XCCSIT,YCCSIT,XDIST,YDIST
0431 997 FORMAT(//,A,BET,XCCSIT,YCCSIT,XDIST,YDIST,'.5F8.1)
0432 D08 = SIGNCL * EXPZ * EXP(-YCCSIT * YCCSIT / (2.0 * SIGY * SIGY *
0433 (TWOFI * SIGY * AVSFD * SQRT(0.5 * SQSIGZ)))
0434 CONC = D08 * AVSFD / (SQRTPI * SICK)
0435 CNMPL(I) = CONC
0436 WRITE(LUPRINT,991)AVDIR,AVDIRR,BETAF,ARG1,XDEL,YDEL,XDIST,YDIST,
0437 XSITEP,YSITEP,XSTAB,YSTAB
0438 991 FORMAT('5X"AVDIR",4X"AVDIRR",5X"BETAF",6X"ARG1",6X"XDEL",
0439 6X"YDEL",5X"XDIST",5X"YDIST",4X"XSITEP",4X"YSITEP",
0440 5X"XSTAB",5X"YSTAB"
0441 /12F10.1)
0442 74 DNMPL(I) = D08
0443 C
0444 C CALCULATE AND WRITE OUT THE CONCENTRATION AND DOSAGE FOR
0445 C EACH MATERIAL
0446 C
0447 DELC = ABS(0.5 * (2.0 * CNMPL(I) - CNMPL(2) - CNMPL(3)))
0448 DELD = ABS(0.5 * (2.0 * DNMPL(I) - DNMPL(2) - DNMPL(3)))
0449 WRITE (LUPRINT,207) <MATS(I,1),I=1,3>,CNMPL(I),DELC,DMNPL(I),DELD
0450 C
0451 ARG1 = PCO/PHCL
0452 CONC = ARG1 * CNMPL(I)
0453 DLC = ARG1 * DELC
0454 D0S = ARG1 * DNMPL(I)
0455 DLD = ARG1 * DELD
0456 WRITE (LUPRINT,207) <MATS(I,2),I=1,3>,CONC,DLC,D0S,DLD
0457 C
0458 ARG1 = PCO2/PHCL
0459 CONC = ARG1 * CNMPL(I)
0460 DLC = ARG1 * DELC
0461 D0S = ARG1 * DNMPL(I)
0462 DLD = ARG1 * DELD
0463 WRITE (LUPRINT,207) <MATS(I,3),I=1,3>,CONC,DLC,D0S,DLD
0464 C
0465 ARG1 = PAL203/PHCL * 0.43882420 * PRESS(I) /
0466 (TEMP(I) + 273.16)
0467 CONC = ARG1 * CNMPL(I)
0468 DLC = ARG1 * DELC
0469 D0S = ARG1 * DNMPL(I)
0470 DLD = ARG1 * DELD
0471 WRITE (LUPRINT,207) <MATS(I,4),I=1,3>,CONC,DLC,D0S,DLD
0472 C
0473 ARG1 = PHO/PHCL
0474 CONC = ARG1 * CNMPL(I)
0475 DLC = ARG1 * DELC
0476 D0S = ARG1 * DNMPL(I)
0477 DLD = ARG1 * DELD
0478 WRITE (LUPRINT,207) <MATS(I,5),I=1,3>,CONC,DLC,D0S,DLD
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```

0479 C          REQUEST ANOTHER POINT FOR AN OFF CENTER CONCENTRATION
0480 C          CALCULATION
0481 C
0482 C
0483 C          77 GO TO 71
0484 C
0485 C          FINISHED WITH OFF CENTER CONCENTRATIONS
0486 C
0487 C          78 YPOS = YPOS - 32.0
0488 C
0489 C          IS AN ISOPLETH PLOT DESIRED?
0490 C
0491 C          61 CALL DREAD(NAMEF,4,ILINE)
0492 C          CALL LERS(YPOS,LVERSN)
0493 C          CALL CHAR(0.,YPOS,0,ILINE,62,3,0)
0494 C          CALL CHAR(384.,YPOS,0,ILINE(5),8,3,0)
0495 C          CALL CHAR(464.,YPOS,0,ILINE(50),6,0,0)
0496 C
0497 C          IF PLASMASCOPE VERSION, SKIP TO THE CALL TO "IN"
0498 C
0499 C          IF(LVERSN) GO TO 812
0500 C
0501 C          OTHERWISE RESPOND WITH Y FOR YES AND N FOR NO TO ANSWER
0502 C          QUESTIONS.
0503 C
0504 C          READ(LU,90) IANS
0505 C          IF(IANS.EQ.1H0)STOP
0506 C
0507 C          "YES" IS THE DEFAULT CASE
0508 C
0509 C          IX = 25
0510 C
0511 C          THIS IS THE "NO". RESPONSE
0512 C
0513 C          IF(IANS.NE.NO) GO TO 813
0514 C          IX = 30
0515 C          GO TO 813
0516 C          CONTINUE
0517 C          CALL IN(1,JTYPE,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.)
0518 C          CONTINUE
0519 C          CALL CHAR(0.,YPOS,0,ILINE,64,0,0)
0520 C          IF(IX.LE.25)CALL CHAR(464.,YPOS,0,IERS,6,0,0)
0521 C          IF(IX.GT.25)CALL CHAR(384.,YPOS,0,IERS,8,0,0)
0522 C          YPOS = YPOS - 32.
0523 C
0524 C          IF AN ISOPLETH PLOT IS DESIRED, CALL THE PROGRAM RISOP
0525 C
0526 C          IF(IX.GE.28)GO TO 87
0527 C          CALL MGRAF
0528 C          CALL RWDIS(NAME,1)
0529 C          CALL EXEC(9,RISOP)
0530 C          CALL RWDIS(NAME,0)
0531 C          CALL GRAF(1)
0532 C          87 CONTINUE
0533 C
0534 C          CALL MGRAF TO RETURN SCOPE TO NORMAL MODE OF OPERATION
0535 C          (APPROPRIATE ONLY WHEN USING PLASMASCOPE)
0536 C
0537 C          86 CALL MGRAF
0538 C

```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0539 C WRITE OUT THE COMMON DISK FILE  
0540 C  
0541 C CALL RUDIS(NAME,1)  
0542 C  
0543 C RETURN TO THE MAIN PROGRAM REEDS  
0544 C  
0545 C STOP  
0546 C  
0547 C *END  
0548 C END*
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
ARCOUT T=00003 IS ON CR00003 USING 00052 BLKS R=0000
0001 FTM4,L
0002 SUBROUTINE RUDIS(NAME,J), ...UPDATED 12 22 79, C. BORK (EDJ)
0003 C
0004 C
0005 C
0006 C - THIS SUBROUTINE READS AND WRITES COMMON AREAS TO DISC
0007 C - TO BE UTILIZED BY THE VARIOUS REED PROGRAMS.
0008 C
0009 C
0010 C
0011 C >>> COMMON AREAS
0012 C
0013 C
0014 C >>> MATH PARAMETERS
0015 C COMMON PI,P10VR2,P143,TU0P1,SQR2PI,DEGRAD,RADDEG,IV2
0016 C
0017 C >>> VEHICLE PARAMETERS
0018 C COMMON VPAR(18),SIGHL
0019 C
0020 C >>> OPTION PARAMETERS
0021 C COMMON YPOS,YES,ISMOFO,IVHICL,IRUN,ICALC,IPLACE,ITIMEZ,IGOTMP,
0022 C HUMRUN,LU,NUM,MOHLY,MFORH,LUPRNT,ITDU,IPL1,IPL2,IPL3
0023 C INTEGER YES
0024 C
0025 C >>> MET DATA
0026 C COMMON ALT(30),IDIR(30),SPEED(30),TEMP(30),PRESS(30),PTEMP(30),
0027 C SURDEN,FILE(3)
0028 C INTEGER FILE
0029 C
0030 C >>> CALCULATION TIME
0031 C COMMON ITIME,IDAY,IMONTH(2),IYEAR
0032 C
0033 C >>> SOUNDING TIME
0034 C COMMON ISTIME,ISDAY,ISMOK(2),ISYEAR
0035 C
0036 C >>> LAUNCH TIME
0037 C COMMON LTIME,LTIM,LDAY,LMONTH(2),LYEAR,LAUNTD(10)
0038 C LOGICAL LTIME
0039 C
0040 C >>> CLOUD STABILIZATION AND LAYER PARAMETERS
0041 C COMMON STBLT,STBTD,STBRZD,STBRNG,STBTIN,CLORAD,RISTIN(30),BOTLAY,
0042 C TOPLAY,CALHT,LAYTOP,LAYBOT,REFLEC,IPTZ
0043 C
0044 C >>> DIFFUSION COEFFICIENTS
0045 C COMMON SIGX0,SIGX,SIGZ0,SIGAP,XDIST,YDIST,EXPZ,SOSIGZ,SIGY,SIGAL,
0046 C CLDNG,AVSFD,AVDIR,SIGZ
0047 C
0048 C >>> CONCENTRATION AND DOSAGE VALUES AND RANGES
0049 C COMMON CONMAX,CONMPK,DOSMAX,DOSPK,INGSNC,INGSND
0050 C
0051 C >>> EQUIVALENCE STATEMENT FOR VEHICLE PARAMETERS
0052 C EQUIVALENCE (Q1,VPAR(1)),(Q2,VPAR(2)),(Q3,VPAR(3)),
0053 C (Q4,VPAR(4)),(Q5,VPAR(5)),(Q6,VPAR(6)),
0054 C (Q7,VPAR(7)),(Q8,VPAR(8)),(Q9,VPAR(9)),
0055 C (HEATN,VPAR(10)),(HEATM,VPAR(11)),(HEATA,VPAR(12)),
0056 C (PHCL,VPAR(13)),(PCO,VPAR(14)),(PCO2,VPAR(15)),
0057 C (PAC,VPAR(16)),(PNO,VPAR(17)),(SPRMAX,VPAR(18))
0058 C INTEGER QDC(14),QDIF(5)
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0059 DIMENSION NAME(3)
0060 EQUIVALENCE (OBUF(1),P1)
0061 CALL OPEN(OUCB,IERR,NAME,0)
0062 IF(IERR.NE.0)CALL CREAT(OUCB,IERR,NAME,14,3,0,0,144)
0063 IF(JJ.EQ.1)CALL WRITF(OUCB,IERR,OBUF,557)
0064 IF(JJ.EQ.0)CALL READF(OUCB,IERR,OBUF,557)
0065 CALL CLOSE(OUCB,IERR)
0066 RETURN
0067 END
0068 SUBROUTINE CPLOT(XDISTV,DOSE,CONCV,NUMY,IEXPD,IEXPC)
0069 C
0070 C
0071 C
0072 C --- THIS SUBROUTINE PLOTS THE DOSAGE AND CONCENTRATION CENTERLINE ---
0073 C --- CURVES ---
0074 C ---
0075 C
0076 C
0077 C >>> COMMON AREAS
0078 C
0079 C
0080 C >>> MATH PARAMETERS
0081 C COMMON P1,PIOVR2,P143,TUOPI,SQR2PI,DEGRAD,RADDEG,IY2
0082 C
0083 C >>> VEHICLE PARAMETERS
0084 C COMMON VPAK(18),SICHEL
0085 C
0086 C >>> OPTION PARAMETERS
0087 C COMMON YPOS,YES,ISWFO,IVHICL,IRUN,ICALC,IPLACE,ITIMEZ,IGOTMP,
0088 C NUMRUN,LU,NUM,MONLY,NFORM,LUPRINT,ITDU,IPL1,IPL2,IPL3
0089 C INTEGER YES
0090 C
0091 C >>> MET DATA
0092 C COMMON ALT(30),IDIR(30),SPEED(30),TEMP(30),PRESS(30),PTENP(30),
0093 C SURDEN,FILE(3)
0094 C INTEGER FILE
0095 C
0096 C >>> CALCULATION TIME
0097 C COMMON ITIME,IDAY,IMONTH(2),IYEAR
0098 C
0099 C >>> SOUNDING TIME
0100 C COMMON ISIME,ISDAY,ISMONK(2),ISYEAR
0101 C
0102 C >>> LAUNCH TIME
0103 C COMMON LTIME,LTIM,LOAD,LMONTH(2),LYEAR,LAUNTD(10)
0104 C LOGICAL LTIME
0105 C
0106 C >>> CLOUD STABILIZATION AND LAYER PARAMETERS
0107 C COMMON STBALT,STBNT,STBAZD,STBRNG,STBTIN,CLRABD,RISTMK(30),BOTLAY,
0108 C TOPLAY,GALHT,LAYTOP,LAYBOT,REFLEC,IPTZ
0109 C
0110 C >>> DIFFUSION COEFFICIENTS
0111 C COMMON SIGX,SIGY,SIGZ0,SIGAP,XDIST,YDIST,EXPZ,SOSIGZ,SIGY,SIGAL,
0112 C CLDNG,AVSPD,AVDIR,SIGAZ
0113 C
0114 C >>> CONCENTRATION AND DOSAGE VALUES AND RANGES
0115 C COMMON CONMAX,CONCPK,DOSEPK,DOSEMAX,DOSEPK,RNGSCH,RNGSMD
0116 C
0117 C >>> EQUIVALENCE STATEMENT FOR VEHICLE PARAMETERS
0118 C
```



ORIGINAL QUALITY

```
0119 EQUIVALENCE (QC1,VPAR(1)),(QC2,VPAR(2)),(QC3,VPAR(3)),
0120 (QT1,VPAR(4)),(QT2,VPAR(5)),(QT3,VPAR(6)),
0121 (QA,VPAR(7)),(QB,VPAR(8)),(CC,VPAR(9)),
0122 (HEATN,VPAR(10)),(HEATM,VPAR(11)),(HEATA,VPAR(12)),
0123 (PHCL,VPAR(13)),(PCO,VPAR(14)),(PCO2,VPAR(15)),
0124 (PAL203,VPAR(16)),(PHO,VPAR(17)),(GAMMAX,VPAR(18))
0125 C
0126 C DIMENSION STATEMENT
0127 C
0128 C DIMENSION XD18TV(1),DOSV(1),CONCV(1)
0129 C
0130 C CALCULATE PLOTTING FACTORS
0131 C
0132 C FXD18T = 9295.0/30000.0
0133 C FDOOS = 8231.0 * 10.0**(IEXPD - 1)
0134 C FCONC = 8231.0 * 10.0**(IEXPC - 1)
0135 C
0136 C PLOT THE DOSAGE CENTERLINE CURVE
0137 C
0138 C DO 7 I=1,NUMV
0139 C J = 1/I
0140 C J = 1 - 2 * J
0141 C IXPLOT = XD18TV(I) * FXD18T + 775.0
0142 C IYPLT = DOSV(I) * FDOOS + 1040.0
0143 C WRITE (IPL2) J,I,IXPLOT,IYPLT
0144 C 7 CALL SYMBL(100,100,254000,IPL2)
0145 C
0146 C PLOT THE CONCENTRATION CENTERLINE CURVE
0147 C
0148 C DO 16 I=1,NUMV
0149 C J = 1/I
0150 C J = 1 - 2 * J
0151 C IXPLOT = XD18TV(I) * FXD18T + 775.0
0152 C IYPLT = CONCV(I) * FCONC + 1040.0
0153 C 16 WRITE (IPL2) J,I,IXPLOT,IYPLT
0154 C
0155 C RETURN TO RCONC
0156 C
0157 C RETURN
0158 C
0159 C END OF CPLOT
0160 C
0161 C
0162 C SUBROUTINE LABEL(IEXPD,IEXPC)
0163 C
0164 C
0165 C
0166 C - THIS SUBROUTINE LABELS THE CONCENTRATION AND DOSAGE CENTERLINE
0167 C - PLOTS
0168 C -
0169 C
0170 C
0171 C >>> COMMON AREAS
0172 C
0173 C
0174 C >>> MATH PARAMETERS
0175 C COMMON PI,P10VR2,P143,IMOP1,SORCF,DEGRAD,RADDEG,IV2
0176 C
0177 C >>> VEHICLE PARAMETERS
0178 C >>>
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```

0179 C COMMON VPAR(18), SIGHEL
0180 C
0181 C >>> OPTION PARAMETERS
0182 C COMMON YFOS, YES, ISMFG, IVHICL, IRUH, ICALC, IPLACE, ITIMEZ, IGOTHP,
0183 C NUMRUM, LU, NUM, NONLY, MFORM, LUPRNT, ITDU, IFL1, IFL2, IFL3
0184 C INTEGER YES
0185 C
0186 C >>> MET DATA
0187 C COMMON ALT(30), IDIR(30), SPEED(30), TEMP(30), PRESS(30), PTEMP(30),
0188 C SURDEN, FILE(3)
0189 C INTEGER FILE
0190 C
0191 C >>> CALCULATION TIME
0192 C COMMON ITIME, IDAY, IMONTH(2), IYEAR
0193 C
0194 C >>> SOUNDING TIME
0195 C COMMON ISIME, ISDAY, ISMONK(2), ISYEAR
0196 C
0197 C >>> LAUNCH TIME
0198 C COMMON LTIME, LTIM, LDAY, LMONTH(2), LYEAR, LAUNTD(10)
0199 C LOGICAL LTIME
0200 C
0201 C >>> CLOUD STABILIZATION AND LAYER PARAMETERS
0202 C COMMON STBAL, STBHT, STBZD, STBRNG, STBTM, CLDRAD, RISTIM(30), BOTLAY,
0203 C TOPLAY, CALHT, LAYTOP, LAYBOT, REFLEC, IPTZ
0204 C
0205 C >>> DIFFUSION COEFFICIENTS
0206 C COMMON SIGX, SIGZ, SIGZ0, SIGAP, XDIST, YDIST, EXPZ, SQSIGZ, SIGY, SIGAL,
0207 C CLDNG, AVSPD, AVDIR, SIGAZ
0208 C
0209 C >>> CONCENTRATION AND DOSAGE VALUES AND RANGES
0210 C COMMON CONMX, CONCPK, DOSMAX, DOSPK, RNGSMC, RNGSMD
0211 C
0212 C >>> EQUIVALENCE STATEMENT FOR VEHICLE PARAMETERS
0213 C EQUIVALENCE (QC1, VPAR(1)), (QC2, VPAR(2)), (QC3, VPAR(3)),
0214 C (QT1, VPAR(4)), (QT2, VPAR(5)), (QT3, VPAR(6)),
0215 C (QA, VPAR(7)), (QB, VPAR(8)), (CC, VPAR(9)),
0216 C (HEATN, VPAR(10)), (HEATA, VPAR(11)), (HEATA, VPAR(12)),
0217 C (PHCL, VPAR(13)), (PCO, VPAR(14)), (PCO2, VPAR(15)),
0218 C (PAL203, VPAR(16)), (PNO, VPAR(17)), (GAMMAX, VPAR(18))
0219 C
0220 C
0221 C
0222 C
0223 C
0224 C
0225 C
0226 C
0227 C
0228 C
0229 C
0230 C
0231 C
0232 C
0233 C
0234 C
0235 C
0236 C
0237 C
0238 C
0239 C
0240 C
0241 C
0242 C
0243 C
0244 C
0245 C
0246 C
0247 C
0248 C
0249 C
0250 C
0251 C
0252 C
0253 C
0254 C
0255 C
0256 C
0257 C
0258 C
0259 C
0260 C
0261 C
0262 C
0263 C
0264 C
0265 C
0266 C
0267 C
0268 C
0269 C
0270 C
0271 C
0272 C
0273 C
0274 C
0275 C
0276 C
0277 C
0278 C
0279 C
0280 C
0281 C
0282 C
0283 C
0284 C
0285 C
0286 C
0287 C
0288 C
0289 C
0290 C
0291 C
0292 C
0293 C
0294 C
0295 C
0296 C
0297 C
0298 C
0299 C
0300 C
0301 C
0302 C
0303 C
0304 C
0305 C
0306 C
0307 C
0308 C
0309 C
0310 C
0311 C
0312 C
0313 C
0314 C
0315 C
0316 C
0317 C
0318 C
0319 C
0320 C
0321 C
0322 C
0323 C
0324 C
0325 C
0326 C
0327 C
0328 C
0329 C
0330 C
0331 C
0332 C
0333 C
0334 C
0335 C
0336 C
0337 C
0338 C
0339 C
0340 C
0341 C
0342 C
0343 C
0344 C
0345 C
0346 C
0347 C
0348 C
0349 C
0350 C
0351 C
0352 C
0353 C
0354 C
0355 C
0356 C
0357 C
0358 C
0359 C
0360 C
0361 C
0362 C
0363 C
0364 C
0365 C
0366 C
0367 C
0368 C
0369 C
0370 C
0371 C
0372 C
0373 C
0374 C
0375 C
0376 C
0377 C
0378 C
0379 C
0380 C
0381 C
0382 C
0383 C
0384 C
0385 C
0386 C
0387 C
0388 C
0389 C
0390 C
0391 C
0392 C
0393 C
0394 C
0395 C
0396 C
0397 C
0398 C
0399 C
0400 C
0401 C
0402 C
0403 C
0404 C
0405 C
0406 C
0407 C
0408 C
0409 C
0410 C
0411 C
0412 C
0413 C
0414 C
0415 C
0416 C
0417 C
0418 C
0419 C
0420 C
0421 C
0422 C
0423 C
0424 C
0425 C
0426 C
0427 C
0428 C
0429 C
0430 C
0431 C
0432 C
0433 C
0434 C
0435 C
0436 C
0437 C
0438 C
0439 C
0440 C
0441 C
0442 C
0443 C
0444 C
0445 C
0446 C
0447 C
0448 C
0449 C
0450 C
0451 C
0452 C
0453 C
0454 C
0455 C
0456 C
0457 C
0458 C
0459 C
0460 C
0461 C
0462 C
0463 C
0464 C
0465 C
0466 C
0467 C
0468 C
0469 C
0470 C
0471 C
0472 C
0473 C
0474 C
0475 C
0476 C
0477 C
0478 C
0479 C
0480 C
0481 C
0482 C
0483 C
0484 C
0485 C
0486 C
0487 C
0488 C
0489 C
0490 C
0491 C
0492 C
0493 C
0494 C
0495 C
0496 C
0497 C
0498 C
0499 C
0500 C
0501 C
0502 C
0503 C
0504 C
0505 C
0506 C
0507 C
0508 C
0509 C
0510 C
0511 C
0512 C
0513 C
0514 C
0515 C
0516 C
0517 C
0518 C
0519 C
0520 C
0521 C
0522 C
0523 C
0524 C
0525 C
0526 C
0527 C
0528 C
0529 C
0530 C
0531 C
0532 C
0533 C
0534 C
0535 C
0536 C
0537 C
0538 C
0539 C
0540 C
0541 C
0542 C
0543 C
0544 C
0545 C
0546 C
0547 C
0548 C
0549 C
0550 C
0551 C
0552 C
0553 C
0554 C
0555 C
0556 C
0557 C
0558 C
0559 C
0560 C
0561 C
0562 C
0563 C
0564 C
0565 C
0566 C
0567 C
0568 C
0569 C
0570 C
0571 C
0572 C
0573 C
0574 C
0575 C
0576 C
0577 C
0578 C
0579 C
0580 C
0581 C
0582 C
0583 C
0584 C
0585 C
0586 C
0587 C
0588 C
0589 C
0590 C
0591 C
0592 C
0593 C
0594 C
0595 C
0596 C
0597 C
0598 C
0599 C
0600 C
0601 C
0602 C
0603 C
0604 C
0605 C
0606 C
0607 C
0608 C
0609 C
0610 C
0611 C
0612 C
0613 C
0614 C
0615 C
0616 C
0617 C
0618 C
0619 C
0620 C
0621 C
0622 C
0623 C
0624 C
0625 C
0626 C
0627 C
0628 C
0629 C
0630 C
0631 C
0632 C
0633 C
0634 C
0635 C
0636 C
0637 C
0638 C
0639 C
0640 C
0641 C
0642 C
0643 C
0644 C
0645 C
0646 C
0647 C
0648 C
0649 C
0650 C
0651 C
0652 C
0653 C
0654 C
0655 C
0656 C
0657 C
0658 C
0659 C
0660 C
0661 C
0662 C
0663 C
0664 C
0665 C
0666 C
0667 C
0668 C
0669 C
0670 C
0671 C
0672 C
0673 C
0674 C
0675 C
0676 C
0677 C
0678 C
0679 C
0680 C
0681 C
0682 C
0683 C
0684 C
0685 C
0686 C
0687 C
0688 C
0689 C
0690 C
0691 C
0692 C
0693 C
0694 C
0695 C
0696 C
0697 C
0698 C
0699 C
0700 C
0701 C
0702 C
0703 C
0704 C
0705 C
0706 C
0707 C
0708 C
0709 C
0710 C
0711 C
0712 C
0713 C
0714 C
0715 C
0716 C
0717 C
0718 C
0719 C
0720 C
0721 C
0722 C
0723 C
0724 C
0725 C
0726 C
0727 C
0728 C
0729 C
0730 C
0731 C
0732 C
0733 C
0734 C
0735 C
0736 C
0737 C
0738 C
0739 C
0740 C
0741 C
0742 C
0743 C
0744 C
0745 C
0746 C
0747 C
0748 C
0749 C
0750 C
0751 C
0752 C
0753 C
0754 C
0755 C
0756 C
0757 C
0758 C
0759 C
0760 C
0761 C
0762 C
0763 C
0764 C
0765 C
0766 C
0767 C
0768 C
0769 C
0770 C
0771 C
0772 C
0773 C
0774 C
0775 C
0776 C
0777 C
0778 C
0779 C
0780 C
0781 C
0782 C
0783 C
0784 C
0785 C
0786 C
0787 C
0788 C
0789 C
0790 C
0791 C
0792 C
0793 C
0794 C
0795 C
0796 C
0797 C
0798 C
0799 C
0800 C
0801 C
0802 C
0803 C
0804 C
0805 C
0806 C
0807 C
0808 C
0809 C
0810 C
0811 C
0812 C
0813 C
0814 C
0815 C
0816 C
0817 C
0818 C
0819 C
0820 C
0821 C
0822 C
0823 C
0824 C
0825 C
0826 C
0827 C
0828 C
0829 C
0830 C
0831 C
0832 C
0833 C
0834 C
0835 C
0836 C
0837 C
0838 C
0839 C
0840 C
0841 C
0842 C
0843 C
0844 C
0845 C
0846 C
0847 C
0848 C
0849 C
0850 C
0851 C
0852 C
0853 C
0854 C
0855 C
0856 C
0857 C
0858 C
0859 C
0860 C
0861 C
0862 C
0863 C
0864 C
0865 C
0866 C
0867 C
0868 C
0869 C
0870 C
0871 C
0872 C
0873 C
0874 C
0875 C
0876 C
0877 C
0878 C
0879 C
0880 C
0881 C
0882 C
0883 C
0884 C
0885 C
0886 C
0887 C
0888 C
0889 C
0890 C
0891 C
0892 C
0893 C
0894 C
0895 C
0896 C
0897 C
0898 C
0899 C
0900 C
0901 C
0902 C
0903 C
0904 C
0905 C
0906 C
0907 C
0908 C
0909 C
0910 C
0911 C
0912 C
0913 C
0914 C
0915 C
0916 C
0917 C
0918 C
0919 C
0920 C
0921 C
0922 C
0923 C
0924 C
0925 C
0926 C
0927 C
0928 C
0929 C
0930 C
0931 C
0932 C
0933 C
0934 C
0935 C
0936 C
0937 C
0938 C
0939 C
0940 C
0941 C
0942 C
0943 C
0944 C
0945 C
0946 C
0947 C
0948 C
0949 C
0950 C
0951 C
0952 C
0953 C
0954 C
0955 C
0956 C
0957 C
0958 C
0959 C
0960 C
0961 C
0962 C
0963 C
0964 C
0965 C
0966 C
0967 C
0968 C
0969 C
0970 C
0971 C
0972 C
0973 C
0974 C
0975 C
0976 C
0977 C
0978 C
0979 C
0980 C
0981 C
0982 C
0983 C
0984 C
0985 C
0986 C
0987 C
0988 C
0989 C
0990 C
0991 C
0992 C
0993 C
0994 C
0995 C
0996 C
0997 C
0998 C
0999 C
1000 C

```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0235 WRITE (IPL2) -1.1,3700,8745
0240 WRITE (IPL2,201) 125.0,0.0,125.81BTIM
0241 WRITE (IPL2) -1.1,3700,8540
0242 WRITE (IPL2,202) 125.0,0.0,125.COMMAX
0243 WRITE (IPL2) -1.1,3700,8335
0244 WRITE (IPL2,201) 125.0,0.0,125.TOPLAY
0245 WRITE (IPL2) -1.1,3700,8130
0246 WRITE (IPL2,201) 125.0,0.0,125.80TLAY
0247 WRITE (IPL2) -1.1,3700,7925
0248 WRITE (IPL2,201) 125.0,0.0,125.CALHT
0249 IF<ISHMDFO.EQ.1>GO TO 4
0250 WRITE (IPL2) -1.1,5625,9000
0251 WRITE (IPL2) 1.1,6125,9000
0252 GO TO 7
0253
0254
0255
0256
0257
0258
0259
0260
0261
0262
0263
0264
0265
0266
0267
0268
0269
0270
0271
0272
0273
0274
0275
0276
0277
0278
0279
0280
0281
0282
0283
0284
0285
0286
0287
0288
0289
0290
0291
0292
0293
0294
0295
0296
0297
0298

4
1
10

WRITE (IPL2,203) 125.0,0.0,125.ISTIME,ITIMEZ,LAUNTO<4>,ISDAY,ISMOM,
LMONTH,LYEAR
WRITE (IPL2) -1.1,5725,8745
WRITE (IPL2,204) 125.0,0.0,125.ITIME,LAUNTO<4>,IDAY,IMONTH,IYEAR
IF<LTIME>WRITE<IPL2,205>205.0,0.205,LTIM,LAUNTO<3>,LAUNTO<4>,LDAY,
LMONTH,LYEAR
RETURN TO RCONC
RETURN
END OF LABEL
END
SUBROUTINE DFEXP<J,CONC>
-----
THIS SUBROUTINE CALCULATES DIFFUSION EXPONENTIALS
J - INDEX IN THE ALT ARRAY OF THE TOP OF THE LAYER
CONC - CONCENTRATION TO BE TESTED
-----
>>> COMMON AREAS
>>> MATH PARAMETERS
COMMON PI,PIOV2,P143,TWOPI,SQR2PI,DEGRAD,RADDEG,IV2
>>> VEHICLE PARAMETERS
COMMON VPAR<18>,SIGICL
>>> OPTION PARAMETERS
COMMON YPOS,YES,ISINDFO,IYHICL,IRUN,ICALC,IPLACE,ITIMEZ,IGOTMF,
IRUNUN,LU,NUM,MOHLY,MFORM,LUPRNT,ITDU,IPL1,IPL2,IPL3
INTEGER YES
NET DAT
COMMON ALT<30>,IDIR<30>,SPEED<30>,TEMP<30>,PRES<30>,DPT<30>,
SURDEN,FILE<3>
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```

0299 C INTEGER FILE
0300 C
0301 C >>> CALCULATION TIME
0302 C COMMON ITIME, IDAY, IMONTH(2), IYEAR
0303 C
0304 C >>> SOUNDING TIME
0305 C COMMON ISTEME, ISDAY, ISMONK(2), ISYEAR
0306 C
0307 C >>> LAUNCH TIME
0308 C COMMON LTIME, LTIM, LDAY, LMONTHK(2), LYEAR, LAUNTD(10)
0309 C LOGICAL LTIME
0310 C
0311 C >>> CLOUD STABILIZATION AND LAYER PARAMETERS
0312 C COMMON STBAL, STBHT, STBAZD, STBRNG, STBTIM, CLDRAD, RISTIM(30), BOTLAY,
0313 C TOPLAY, CALHT, LAYTOP, LAYBOT, REFLEC, IPT2
0314 C
0315 C >>> DIFFUSION COEFFICIENTS
0316 C COMMON SIGX0, SIGX, SIGZ0, SIGAP, XDIST, YDIST, EXPZ, SOSIGZ, SIGY, SIGAL,
0317 C TCLNG, AVSPD, AVDIR, SIGAZ
0318 C
0319 C >>> CONCENTRATION AND DOSAGE VALUES AND RANGES
0320 C COMMON CONMAX, CONCPK, DOSMAX, DOSFK, RINGSMC, RINGSMD
0321 C
0322 C >>> EQUIVALENCE STATEMENT FOR VEHICLE PARAMETERS
0323 C EQUIVALENCE (QC1, VPAR(1)), (QC2, VPAR(2)), (QC3, VPAR(3)),
0324 C (QT1, VPAR(4)), (QT2, VPAR(5)), (QT3, VPAR(6)),
0325 C (AA, VPAR(7)), (BB, VPAR(8)), (CC, VPAR(9)),
0326 C (HEATN, VPAR(10)), (HEATA, VPAR(11)), (HEATA, VPAR(12)),
0327 C (PHCL, VPAR(13)), (PCO, VPAR(14)), (PCO2, VPAR(15)),
0328 C (PAL203, VPAR(16)), (PHO, VPAR(17)), (GANNAX, VPAR(18))
0329 C
0330 C
0331 C CALCULATE SIGMA Z
0332 C
0333 C SIGZ = XDIST * SIGAP + SIGZ0/1.20
0334 C SOSIGZ = 2.0 * SIGZ * SIGZ
0335 C
0336 C CALCULATE THE EXPONENTIAL SUM IN THE DIFFUSION EQUATION
0337 C
0338 C
0339 C
0340 C
0341 C
0342 C
0343 C
0344 C
0345 C
0346 C
0347 C
0348 C
0349 C
0350 C
0351 C
0352 C
0353 C
0354 C
0355 C
0356 C
0357 C
0358 C
0359 C
0360 C
0361 C
0362 C
0363 C
0364 C
0365 C
0366 C
0367 C
0368 C
0369 C
0370 C
0371 C
0372 C
0373 C
0374 C
0375 C
0376 C
0377 C
0378 C
0379 C
0380 C
0381 C
0382 C
0383 C
0384 C
0385 C
0386 C
0387 C
0388 C
0389 C
0390 C
0391 C
0392 C
0393 C
0394 C
0395 C
0396 C
0397 C
0398 C
0399 C
0400 C
0401 C
0402 C
0403 C
0404 C
0405 C
0406 C
0407 C
0408 C
0409 C
0410 C
0411 C
0412 C
0413 C
0414 C
0415 C
0416 C
0417 C
0418 C
0419 C
0420 C
0421 C
0422 C
0423 C
0424 C
0425 C
0426 C
0427 C
0428 C
0429 C
0430 C
0431 C
0432 C
0433 C
0434 C
0435 C
0436 C
0437 C
0438 C
0439 C
0440 C
0441 C
0442 C
0443 C
0444 C
0445 C
0446 C
0447 C
0448 C
0449 C
0450 C
0451 C
0452 C
0453 C
0454 C
0455 C
0456 C
0457 C
0458 C
0459 C
0460 C
0461 C
0462 C
0463 C
0464 C
0465 C
0466 C
0467 C
0468 C
0469 C
0470 C
0471 C
0472 C
0473 C
0474 C
0475 C
0476 C
0477 C
0478 C
0479 C
0480 C
0481 C
0482 C
0483 C
0484 C
0485 C
0486 C
0487 C
0488 C
0489 C
0490 C
0491 C
0492 C
0493 C
0494 C
0495 C
0496 C
0497 C
0498 C
0499 C
0500 C
0501 C
0502 C
0503 C
0504 C
0505 C
0506 C
0507 C
0508 C
0509 C
0510 C
0511 C
0512 C
0513 C
0514 C
0515 C
0516 C
0517 C
0518 C
0519 C
0520 C
0521 C
0522 C
0523 C
0524 C
0525 C
0526 C
0527 C
0528 C
0529 C
0530 C
0531 C
0532 C
0533 C
0534 C
0535 C
0536 C
0537 C
0538 C
0539 C
0540 C
0541 C
0542 C
0543 C
0544 C
0545 C
0546 C
0547 C
0548 C
0549 C
0550 C
0551 C
0552 C
0553 C
0554 C
0555 C
0556 C
0557 C
0558 C
0559 C
0560 C
0561 C
0562 C
0563 C
0564 C
0565 C
0566 C
0567 C
0568 C
0569 C
0570 C
0571 C
0572 C
0573 C
0574 C
0575 C
0576 C
0577 C
0578 C
0579 C
0580 C
0581 C
0582 C
0583 C
0584 C
0585 C
0586 C
0587 C
0588 C
0589 C
0590 C
0591 C
0592 C
0593 C
0594 C
0595 C
0596 C
0597 C
0598 C
0599 C
0600 C
0601 C
0602 C
0603 C
0604 C
0605 C
0606 C
0607 C
0608 C
0609 C
0610 C
0611 C
0612 C
0613 C
0614 C
0615 C
0616 C
0617 C
0618 C
0619 C
0620 C
0621 C
0622 C
0623 C
0624 C
0625 C
0626 C
0627 C
0628 C
0629 C
0630 C
0631 C
0632 C
0633 C
0634 C
0635 C
0636 C
0637 C
0638 C
0639 C
0640 C
0641 C
0642 C
0643 C
0644 C
0645 C
0646 C
0647 C
0648 C
0649 C
0650 C
0651 C
0652 C
0653 C
0654 C
0655 C
0656 C
0657 C
0658 C
0659 C
0660 C
0661 C
0662 C
0663 C
0664 C
0665 C
0666 C
0667 C
0668 C
0669 C
0670 C
0671 C
0672 C
0673 C
0674 C
0675 C
0676 C
0677 C
0678 C
0679 C
0680 C
0681 C
0682 C
0683 C
0684 C
0685 C
0686 C
0687 C
0688 C
0689 C
0690 C
0691 C
0692 C
0693 C
0694 C
0695 C
0696 C
0697 C
0698 C
0699 C
0700 C
0701 C
0702 C
0703 C
0704 C
0705 C
0706 C
0707 C
0708 C
0709 C
0710 C
0711 C
0712 C
0713 C
0714 C
0715 C
0716 C
0717 C
0718 C
0719 C
0720 C
0721 C
0722 C
0723 C
0724 C
0725 C
0726 C
0727 C
0728 C
0729 C
0730 C
0731 C
0732 C
0733 C
0734 C
0735 C
0736 C
0737 C
0738 C
0739 C
0740 C
0741 C
0742 C
0743 C
0744 C
0745 C
0746 C
0747 C
0748 C
0749 C
0750 C
0751 C
0752 C
0753 C
0754 C
0755 C
0756 C
0757 C
0758 C
0759 C
0760 C
0761 C
0762 C
0763 C
0764 C
0765 C
0766 C
0767 C
0768 C
0769 C
0770 C
0771 C
0772 C
0773 C
0774 C
0775 C
0776 C
0777 C
0778 C
0779 C
0780 C
0781 C
0782 C
0783 C
0784 C
0785 C
0786 C
0787 C
0788 C
0789 C
0790 C
0791 C
0792 C
0793 C
0794 C
0795 C
0796 C
0797 C
0798 C
0799 C
0800 C
0801 C
0802 C
0803 C
0804 C
0805 C
0806 C
0807 C
0808 C
0809 C
0810 C
0811 C
0812 C
0813 C
0814 C
0815 C
0816 C
0817 C
0818 C
0819 C
0820 C
0821 C
0822 C
0823 C
0824 C
0825 C
0826 C
0827 C
0828 C
0829 C
0830 C
0831 C
0832 C
0833 C
0834 C
0835 C
0836 C
0837 C
0838 C
0839 C
0840 C
0841 C
0842 C
0843 C
0844 C
0845 C
0846 C
0847 C
0848 C
0849 C
0850 C
0851 C
0852 C
0853 C
0854 C
0855 C
0856 C
0857 C
0858 C
0859 C
0860 C
0861 C
0862 C
0863 C
0864 C
0865 C
0866 C
0867 C
0868 C
0869 C
0870 C
0871 C
0872 C
0873 C
0874 C
0875 C
0876 C
0877 C
0878 C
0879 C
0880 C
0881 C
0882 C
0883 C
0884 C
0885 C
0886 C
0887 C
0888 C
0889 C
0890 C
0891 C
0892 C
0893 C
0894 C
0895 C
0896 C
0897 C
0898 C
0899 C
0900 C
0901 C
0902 C
0903 C
0904 C
0905 C
0906 C
0907 C
0908 C
0909 C
0910 C
0911 C
0912 C
0913 C
0914 C
0915 C
0916 C
0917 C
0918 C
0919 C
0920 C
0921 C
0922 C
0923 C
0924 C
0925 C
0926 C
0927 C
0928 C
0929 C
0930 C
0931 C
0932 C
0933 C
0934 C
0935 C
0936 C
0937 C
0938 C
0939 C
0940 C
0941 C
0942 C
0943 C
0944 C
0945 C
0946 C
0947 C
0948 C
0949 C
0950 C
0951 C
0952 C
0953 C
0954 C
0955 C
0956 C
0957 C
0958 C
0959 C
0960 C
0961 C
0962 C
0963 C
0964 C
0965 C
0966 C
0967 C
0968 C
0969 C
0970 C
0971 C
0972 C
0973 C
0974 C
0975 C
0976 C
0977 C
0978 C
0979 C
0980 C
0981 C
0982 C
0983 C
0984 C
0985 C
0986 C
0987 C
0988 C
0989 C
0990 C
0991 C
0992 C
0993 C
0994 C
0995 C
0996 C
0997 C
0998 C
0999 C
1000 C

```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0359 C          CALCULATE SIGMA Y
0360 C
0361 CC
0362 CC**THE DIFFERENCE IN THE WIND DIRECTIONS USED IN THE ALGORITHM
0363 CC**FOR STANDARD DEVIATION OF THE LATERAL CONCENTRATION MUST BE
0364 CC**LESS THAN 180 DEGREES...THE FOLLOWING TWO LINES OF CODE
0365 CC**HAVE BEEN ADDED
0366 ITHET = IABS(IDIR(J) - IDIR(I))
0367 IF(ITHET.GE.180) ITHET = 360 - ITHET
0368 SIGAL = XDIST * SIGAP + SIGX0
0369 SIGY = SORT(SIGAL * SIGAL +
0370           (0.0040589 * FLOAT(ITHET) * XDIST)**2)
0371 C
0372 C          CALCULATE CLOUD LENGTH
0373 C
0374 DELVEL = SPEED(J) - SPEED(I)
0375 CLOUDG = 0.29 * DELVEL * XDIST/AVSPD
0376 IF(DELVEL .GE. 0.0)GO TO 11
0377 IF(PTEMP(J)-PTEMP(I) .GT. 0.0)CLOUDG = 0.0
0378 CLOUDG = ABS(CLOUDG)
0379 C
0380 C          CALCULATE SIGMA X
0381 C
0382 I1 SIGX = SORT((CLOUDG/4.3)**2 + SIGX0 * SIGX0)
0383 C
0384 C*** CALCULATE TOTAL CLOUD LENGTH...THIS PARAMETER REPLACES CLOUDG IN COMMON
0385 C*** CRANER DOCUMENTATION IN ERROR...SEE STEPHENS/BJORK/HIPPS PERSONAL COPY
0386 TCLNG = 4.3 * SIGX
0387 C
0388 C          IF CONC=1000.0, DO NOT CALCULATE CROSS WIND DISTANCE BUT RETURN
0389 C          TO THE CALLING PROGRAM
0390 C
0391 IF(CONC .EQ. 1000.0)GO TO 20
0392 C
0393 C          CALCULATE CROSS WIND DISTANCE
0394 C
0395 YDIST = - 2.0 * SIGY * SIGY + ALOG(15.7496 * CONC * SIGX * SIGY *
0396           SIGZ/(SIGHCL * EXPZ))
0397 YDIST = SORT(AMAXI(YDIST,0.0))
0398 C
0399 C          RETURN TO THE CALLING PROGRAM
0400 C
0401 C          CONTINUE
0402 C          WRITE(6,98)
0403 C          FORMAT(6X,"TL",1X,"DELVEL",6X,"CONC",7X,"AVSPD",7X,"SIGAL",
0404 C          7X,"SIGAP",7X,"SIGX0",2X,"J",3X,"PT(J)",3X,"PT(I)",
0405 C          3X,"TEMP2",3X,"TEMP3")
0406 C          WRITE(6,99) TCLNG,DELVEL,CONC,AVSPD,SIGAL,SIGAP,SIGX0,J,PTEMP(J),
0407 C          PTEMP(I),TEMP2,TEMP3
0408 C          FORMAT(1X,0F8.1,F6.2,1P3E12.4,2E12.4,13,0P4F8.2)
0409 C          RETURN
0410 C
0411 C          END OF DFEXP
0412 C
0413 C          SUBROUTINE ORGINIX0,IY0,LVERSN)
0414 C
0415 C
0416 C
0417 C
0418 C          THIS SUBROUTINE GIVES THE APPROPRIATE COORDINATES FOR PLOTTING
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```

FOR THE COMPLEX AND MAP SELECTED
-----
0419 C >>> COMMON AREAS
0420 C
0421 C
0422 C
0423 C
0424 C >>> MATH PARAMETERS
0425 C
0426 C
0427 C COMMON P1,PIOVK2,P143,TWOPI,SOR2PI,DEGRAD,RADEG,IV2
0428 C
0429 C >>> VEHICLE PARAMETERS9
0430 C
0431 C COMMON VPAK(18),SIGHCL
0432 C
0433 C >>> OPTION PARAMETERS
0434 C COMMON YPOS,YES,ISMOFO,IVHICL,IRUN,ICALC,IPLACE,ITIMEZ,IGOTHF,
0435 C NUMRUN,LU,NUM,MONLY,MFORM,LUFRT,ITDU,IPL1,IPL2,IPL3
0436 C INTEGER YES
0437 C
0438 C >>> MET DATA
0439 C COMMON ALT(30),IDIR(30),SPEED(30),TEMP(30),PRESS(30),PTEMP(30),
0440 C SURDEN,FILE(3)
0441 C
0442 C >>> CALCULATION TIME
0443 C COMMON ITIME,IDAY,IMONTH(2),IYEAR
0444 C
0445 C >>> SOUNDING TIME
0446 C
0447 C COMMON ISTIME,ISDAY,ISMONK(2),ISYEAR
0448 C
0449 C >>> LAUNCH TIME
0450 C COMMON LTIME,LTIM,LDAY,LMONTH(2),LYEAR,LAUNTD(10)
0451 C LOGICAL LTIME
0452 C
0453 C >>> CLOUD ABILIZATION AND LAYER PARAMETERS
0454 C COMMON STBAL,STBHT,STBAZD,STBRNG,STBTM,CLDRAD,RISTIM(30),BOTLAY,
0455 C TOPLAY,CALHT,LAYTOP,LAYBOT,REFLEC,IPTZ
0456 C
0457 C >>> DIFFUSION COEFFICIENTS
0458 C COMMON SIGX0,SIGX,SIGZ0,SIGAP,XDIST,YDIST,EXPZ,SQSIGZ,SIGY,SIGAL,
0459 C CLDLNG,AV5PD,AVDIR,SIGAZ
0460 C
0461 C >>> CONCENTRATION AND DOSAGE VALUES AND RANGES
0462 C COMMON CONMAX,CONCPK,DO5MAX,DO5PK,RHCSMC,RHCSMD
0463 C
0464 C >>> EQUIVALENCE STATEMENT FOR VEHICLE PARAMETERS
0465 C EQUIVALENCE (OC1,VPAK(1)),(OC2,VPAK(2)),(OC3,VPAK(3)),
0466 C (QT1,VPAK(4)),(QT2,VPAK(5)),(QT3,VPAK(6)),
0467 C (AA,VPAK(7)),(BB,VPAK(8)),(CC,VPAK(9)),
0468 C (HEATH,VPAK(10)),(HEATH,VPAK(11)),(HEATA,VPAK(12)),
0469 C (PINC,VPAK(13)),(PCG,VPAK(14)),(PCG2,VPAK(15)),
0470 C (PAL203,VPAK(16)),(PHO,VPAK(17)),(GANNAX,VPAK(18))
0471 C DIMENSION ILINE(32),IDATAF(10),IERS(32),IMAPL(48),INMEF(3)
0472 C
0473 C INPUT FORMAT STATEMENT
0474 C
0475 C 100 FORMAT (12,1X,1)
0476 C
0477 C OUTPUT FORMAT STATEMENT
0478 C

```



ORIGINAL PAGE IS  
OF POOR QUALITY.

```
0539 C
0540 C
0541 C
0542 C
0543 C
0544 C
0545 C
0546 C
0547 C
0548 C
0549 C
0550 C
0551 C
0552 C
0553 C
0554 C
0555 C
0556 C
0557 C
0558 C
0559 C
0560 C
0561 C
0562 C
0563 C
0564 C
0565 C
0566 C
0567 C
0568 C
0569 C
0570 C
0571 C
0572 C
0573 C
0574 C
0575 C
0576 C
0577 C
0578 C
0579 C
0580 C
0581 C
0582 C
0583 C
0584 C
0585 C
0586 C
0587 C
0588 C
0589 C
0590 C
0591 C
0592 C
0593 C
0594 C
0595 C
0596 C
0597 C

6 I GOTMP = I
7 CONTINUE
I = IGOTMP
IX0 = IX(I)
IY0 = IY(I)

RETURN TO THE CALLING PROGRAM
RETURN
END OF ORGIN
END
SUBROUTINE SYMBL<INIDE,IHI,ISYMB,LUPLT>
-----
- THIS SUBROUTINE PLOTS A GIVEN SYMBOL A SPECIFIED WIDTH AND
- HEIGHT.
-----
IX=-INIDE/2
IY=-IHI/2
WRITE<LUPLT> -1,-1,IX,IY
WRITE<LUPLT,100> INIDE,0.0,IHI,ISYMB
FORMAT<4I5,A1,1H_>
IY=-IY
WRITE<LUPLT>-1,-1,IX,IY
RETURN
END
SUBROUTINE DREAD<NAMEF,LNUM,ILINE>
-----
- THIS SUBROUTINE READS A SPECIFIED QUESTION FROM A SPECIFIED
- QUESTION FILE. THE QUESTION IS THEN DISPLAYED UPON THE
- PLASMASCOPE FOR PROCESSING.
-----
DIMENSION NAMEF<3>,IDCB<141>,IBUF<40>,ILINE<32>
CALL OPEN<IDCB,IERR,NAMEF,0>
LOOP = LNUM - 1
DO 10 I=1,LOOP
CALL BLANK<IBUF,40>
CALL READF<IDCB,IERR,IBUF>
CONTINUE
CALL BLANK<IBUF,40>
CALL READF<IDCB,IERR,IBUF>
CALL CODE
CALL<IBUF,100> <ILINE<1>,I=1,32>
FORMAT<32A2>
CALL CLOSE<IDCB,IERR>
RETURN
END
SUBROUTINE BLANK<IBUF,II>
```



ORIGINAL PAGE IS  
OF POOR QUALITY

```
0599 C - THIS SUBROUTINE FILLS A SPECIFIED ARRAY WITH A SPECIFIED  
0600 C - NUMBER OF BLANKS.  
0601 C -  
0602 C -  
0603 C -  
0604 C -  
0605 C  
0606 C DIMENSION IBUF(40)  
0607 C DATA IBLK/2H /  
0608 C DO 10 I=1,11  
0609 C IBUF(I) = IALK  
0610 C RETURN  
0611 C END  
0612 C SUBROUTINE LERS(YPOS,LVERSN)  
0613 C -  
0614 C -  
0615 C - THIS SUBROUTINE ERASES A SPECIFIED LINE OF 64-CHARACTERS  
0616 C - FROM THE PLASMASCOPE.  
0617 C -  
0618 C -  
0619 C -  
0620 C DIMENSION IERS(32)  
0621 C DATA IERS/32*2H /  
0622 C IF(YPOS.GT. 48.0)GO TO 4  
0623 C IF(LVERSN.EQ.0)GO TO 4  
0624 C YPOS = 458.0  
0625 C CALL CLEAR  
0626 C RETURN  
0627 C 4 CALL CHAR(0.0,YPOS,0,IERS,64,0,0)  
0628 C CALL CHAR(0.0,YPOS-16.0,0,IERS,64,0,0)  
0629 C RETURN  
0630 C END
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```

CRIS05 T=00003 IS ON CR00003 USING 00066 BLKS R=0000
FIN4.L
PROGRAM RIS0S(3),REEDS ISOPLETH PLOTTING PROGRAM --- 2/3/79
*****
0001 C
0002 C
0003 C
0004 C
0005 C
0006 C *
0007 C * ISOPLETH PLOTTING PROGRAM -- A PROGRAM IN THE REEDS SERIES *
0008 C * OF PROGRAMS *
0009 C *
0010 C
0011 C
0012 C >>> COMMON AREAS
0013 C
0014 C
0015 C >>> MATH PARAMETERS
0016 C COMMON P1,FI0VR2,P143,TU0P1,SQR2P1,DEGRAD,RAD0EG,IV2
0017 C >>> VEHICLE PARAMETERS
0018 C COMMON VPAK(18),SICHEL
0019 C
0020 C
0021 C >>> OPTION PARAMETERS
0022 C COMMON YPOS,YES,ISMOF0,IVHICL,IRUN,ICALC,IPLACE,ITIMEZ,IGOTHF,
0023 C NUMRUH,LU,NUM,ONLY,NFORM,LUPRNT,ITDU,IPL1,IPL2,IPL3
0024 C INTEGER YES
0025 C
0026 C >>> MET DATA
0027 C COMMON ALT(30),DIR(30),SPEED(30),TEMP(30),PRESS(30),PTEMP(30),
0028 C SURDEN,FILE(3)
0029 C INTEGER FILE
0030 C
0031 C >>> CALCULATION TIME
0032 C COMMON ITIME,IDAY,IMONTH(2),IYEAR
0033 C
0034 C >>> SOUNDING TIME
0035 C COMMON ISTIME,ISDAY,ISMONK(2),ISYEAR
0036 C
0037 C >>> LAUNCH TIME
0038 C COMMON LTIME,LTIM,LDAY,LMONTH(2),LYEAR,LAUNTD(10)
0039 C LOGICAL LTIME, LVERSH
0040 C
0041 C >>> CLOUD STABILIZATION AND LAYER PARAMETERS
0042 C COMMON STBAL,STBHT,STDAZ0,STBRHG,STBTIM,CLURAD,RISTIN(30),BOTLAY,
0043 C TOPLAY,CALHT,LAYTOP,LAYBOT,REFLEC,IFTZ
0044 C
0045 C >>> DIFFUSION COEFFICIENTS
0046 C COMMON SIGX,SIGX,SIGZ0,SIGAF,XDISF,YDISF,EXPZ,SOSIGZ,SIGY,SIGAL,
0047 C CLDNG,MYSFD,AVDIR,SIGAZ
0048 C
0049 C >>> CONCENTRATION AND DOSAGE VALUES AND RANGES
0050 C COMMON CONMAX,CONCPK,DOSMAX,DOSPK,RINGSMC,RINGSND
0051 C
0052 C >>> EQUIVALENCE STATEMENT FOR VEHICLE PARAMETERS
0053 C EQUIVALENCE (QC1,VPAK(1)),(QC2,VPAK(2)),(QC3,VPAK(3)),
0054 C (QT1,VPAK(4)),(QT2,VPAK(5)),(QT3,VPAK(6)),
0055 C (CA,VPAK(7)),(CB,VPAK(8)),(CC,VPAK(9)),
0056 C (CHEAT1,VPAK(10)),(CHEAT2,VPAK(11)),(CHEAT3,VPAK(12)),
0057 C (PAC1,VPAK(13)),(PAC2,VPAK(14)),(PAC3,VPAK(15)),
0058 C (PAC4,VPAK(16)),(PAC5,VPAK(17)),(PAC6,VPAK(18))

```

ORIGINAL PAGE IS  
OF POOR QUALITY

```

0059 C      OUTPUT FORMAT STATEMENTS
0060 C
0061 C      199  FORMAT(A1)
0062 C      200  FORMAT ("1"20X"CLOUD LOCATION AND DIMENSIONS"/
0063 C      7X"TIME FROM LAUNCH"12X"RANGE"5X"AZIMUTH"
0064 C      8X"DIAMETERS (METERS)"/
0065 C      6X"(MIN)"9X"(SEC)"9X"(METERS)"4X"(DEG)"6X"CROSS WIND"
0066 C      4X"ALONG WIND")
0067 C
0068 C      201  FORMAT (7X13,10XF5.1,9XF5.1,4XF5.1,7XF7.1,7XF7.1)
0069 C      202  FORMAT (4I5,14" C"42,2X12,1X42,A1,1X14)
0070 C      203  FORMAT (4I5,A1)
0071 C      204  FORMAT (4I5,F5.2" ")
0072 C      205  FORMAT (4I5" ", "F5.2" ")
0073 C      206  FORMAT (F5.3)
0074 C      207  FORMAT (11)
0075 C      208  FORMAT (4I5,14,1X11,A2,2X12,1X42,A1,1X14)
0076 C
0077 C      TYPE AND DIMENSION STATEMENTS
0078 C
0079 C      LOGICAL DFALTC
0080 C      DIMENSION COMC(10),NAME(3),NAMEF(3),ILINE(32),IDATAF(10),
0081 C      DATA IERS(32),IXA(100),IYA(100),IXB(100),IYB(100)
0082 C      DATA NAME/2H=R,2HEE,2HD2/, NAMEF/2H?R,2HIS,2HOS/,
0083 C      DATA IERS/32*2H /
0084 C
0085 C      CALL GRAF TO INITIALIZE SCOPE (APPROPRIATE ONLY WHEN USING
0086 C      PLOSMASCOPE)
0087 C
0088 C      CALL GRAF(1)
0089 C
0090 C      FIND OUT IF THIS IS THE CRT OR PLOSMASCOPE VERSION OF REED2
0091 C      BEING RUN
0092 C
0093 C      CALL VERSK(1)
0094 C      LVERSH = I.EQ. 0
0095 C
0096 C      READ COMMON DISK FILE
0097 C
0098 C      CALL RWDIS(NAME,0)
0099 C
0100 C      DETERMINE THE ORIGIN ON THE MAP FOR THIS PLOT AND MOVE THE
0101 C      PEN THERE
0102 C
0103 C      CALL ORGIN(X0,IY0,LVERSH)
0104 C
0105 C      WRITE (IPL3) -1,1,IX0,IY0
0106 C
0107 C      DETERMINE THE INDEX IN THE ALTITUDE DATA ARRAY THAT HAS
0108 C      THAT ALTITUDE JUST LOWER THAN THE EFFECTIVE CLOUD HEIGHT, STBIT
0109 C
0110 C      DO 4 I=2,LAYTOP
0111 C      IF(STBIT.GT. ALT(I))GO TO 4
0112 C      ISBIT = I - 1
0113 C      GO TO 5
0114 C      4 CONTINUE
0115 C      5 CONTINUE
0116 C
0117 C      END
0118 C

```

ORIGINAL PAGE IS  
OF POOR QUALITY

```

0119 C      5 X = 0.0
0120 C      Y = 0.0
0121 C      DO 9 I=2,ISTBHT
0122 C      IMI = I - 1
0123 C      RANGE = 0.5 * (RISTIM(I) - RISTIM(IMI)) * (SPEED(I) + SPEED(IMI))
0124 C      DIR = 0.5 * FLOAT(IDIR(I) + IDIR(IMI))
0125 C      DIR = 0.5 * FLOAT(IDIR(I) + IDIR(IMI)) - 180. * (IABS( IDIR(I)
0126 C      - IDIR(IMI)))/180)
0127 C      IF(DIR .LE. 0) DIR = DIR + 360
0128 C      DIR = DEGRAD*DIR
0129 C      IFLIP = 1
0130 C      IF(DIR .GT. PI) IFLIP = -1
0131 C      DIR = DIR + IFLIP*PI
0132 C      X = X + RANGE * COS(PI - DIR)
0133 C      Y = Y + RANGE * SIN(PI - DIR)
0134 C      R = SORT(X*X + Y*Y)
0135 C      IF(IABS(IDIR(I) - IDIR(IMI)) .GT. 180)DIR = DIR - 180.0
0136 C      DIR = 180. - DIR
0137 C      IF(DIR .LT. 0.0)DIR = DIR + 360.0
0138 C      DIR = DEGRAD * (360.0 - DIR)
0139 C      DIR = DEGRAD * DIR
0140 C      X = X + RANGE * COS(DIR)
0141 C      Y = Y + RANGE * SIN(DIR)
0142 C      IX = INT(0.2631 * X) + IX0
0143 C      IY = INT(0.3545 * Y) + IY0
0144 C      IF(IX.LT.0 .OR. IX.GT.9999 .OR. IY.LT.0 .OR. IY.GT.9999)GO TO 11
0145 C      WRITE (IPL3) I, IX, IY
0146 C
0147 C      MAKE THE CALCULATIONS NECESSARY TO WRITE OUT THE CLOUD
0148 C      LOCATION AND DIMENSIONS
0149 C
0150 C      11 ALTI = 0.5 * (STBHT + ALT(ISTBHT))
0151 C      ISTBPI = ISTBHT + 1
0152 C      ARG1 = ALT(ISTBPI) - ALT(ISTBHT)
0153 C      ARG2 = (STBHT - ALT(ISTBHT))/ARG1
0154 C      SPCEHT = SPEED(ISTBHT) + (SPEED(ISTBPI) - SPEED(ISTBHT)) * ARG2
0155 C
0156 C
0157 C      **SINCE THE EFFECTIVE CLOUD HEIGHT IS ONLY AN ARTIFICE BASED ON
0158 C      **THE STABILIZATION HEIGHT, THE TIME TAKEN TO REACH THE EFFECTIVE
0159 C      **CLOUD HEIGHT IS THE TOTAL RISE TIME, NOT A FRACTION OF THE
0160 C      **RISE TIME...THE CODE HAS BEEN CHANGED ACCORDINGLY AFTER CONSUL-
0161 C      **TATION WITH DR. STEPHENS...JSH
0162 C
0163 C      RANGE = SPCEHT * (RISTIM(ISTBPI) - RISTIM(ISTBHT))
0164 C      IF(IABS(IDIR(ISTBPI) - IDIR(ISTBHT)) .LT. 180)GO TO 14
0165 C      IF(IDIR(ISTBPI) .LT. 180)IDIR(ISTBPI) = IDIR(ISTBPI) + 360
0166 C      IF(IDIR(ISTBHT) .LT. 180)IDIR(ISTBHT) = IDIR(ISTBHT) + 360
0167 C      DIR = FLOAT(IDIR(ISTBHT)) + (ALTI - ALT(ISTBHT)) *
0168 C      FLOAT(IDIR(ISTBPI) - IDIR(ISTBHT))/ARG1
0169 C      DIR = FLOAT(IDIR(ISTBHT)) + ((STBHT - ALT(ISTBHT))/ARG1)
0170 C      * FLOAT(IDIR(ISTBPI) - IDIR(ISTBHT))
0171 C      IF(DIR .GT. 360.0)DIR = DIR - 360.0
0172 C      IF(DIR .GT. 180.0)GO TO 17
0173 C      DIR = DIR + 180.0
0174 C      GO TO 18
0175 C      DIR = DIR - 180.0
0176 C      DIR = 180.0 - DIR
0177 C      IF(DIR .GT. 360.0)DIR = DIR - 360.0
0178 C
0179 C      **P = ...

```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0179 IF<DIR.GT.180.0> FLIP = -1.0
0180 DIR = DIR + FLIP * 180.0
0181 ARG1 = DEGRAD * DIR
0182 X = X + RANGE * COS<PI - ARG1>
0183 Y = Y + RANGE * SIN<PI - ARG1>
0184 R = SORT<X * X + Y * Y>
0185 DELR = 300.0 * AVSPD
0186 DACR = 4.30 * SIGX0
0187 DALNG = 4.30 * SIGX0
0188
0189 ANGL = ATAN<Y,X>
0190 ANGL = ANGL * RAINDEC
0191 ARG1 = 180.0
0192 IF<ANGL.GT.180.0> ARG1 = 540.0
0193 AZ = ARG1 - ANGL
0194
0195 CCC
0196 CCC IF<AVDIR.GT.180.0> ARG1 = 540.0
0197 CCC DAZ = ARG1 - AVDIR
0198 CCC ARG1 = DEGRAD * DAZ
0199 IF<AVDIR.GE.90.0> DAZ = 180.0 - AVDIR
0200 IF<AVDIR.GE.180.0> DAZ = AVDIR - 180.0
0201 IF<AVDIR.GE.270.0> DAZ = 360.0 - AVDIR
0202 ARG1 = DEGRAD * DAZ
0203 DELX = DELR * COS<ARG1>
0204 DELY = DELR * SIN<ARG1>
0205
0206 C INITIALIZE SOME QUANTITIES FOR THE SUBSEQUENT DO LOOP
0207 C
0208 IDELTH = IABS<DIR<LAYTOP> - IDIR<1>>
0209 IF<IDELTH.GE.180> IDELTH = 360 - IDELTH
0210 C
0211 DELU = ABS<SPEED<1>STBT> - SPEED<1>>
0212 C
0213 TIM = STBTIM
0214 ITIM = INT<TIM>/60
0215 RE = ANOD<TIM,60.0>
0216 XDIST = 0.0
0217 XC = X
0218 YC = Y
0219 TXL = 0.28 * DELU/AVSPD
0220 SIGX02 = SIGX0 * SIGX0
0221 SIGNAL2 = SIGNAL * SIGNAL
0222 WRITE <LUPRINT,200>
0223 CC
0224 CC*** IN THE FOLLOWING DO LOOP, STBRNG IS THE DISTANCE FROM THE
0225 CC*** PAD TO STABILIZATION, AND "R" IS THE DISTANCE FROM THE PAD
0226 CC*** XDIST IS THE DISTANCE FROM THE STABILIZATION POINT.
0227 CC
0228 DO 22 I=1,13
0229 WRITE (6,666) XDIST
0230 666 FORMAT(' THE DISTANCE FROM STABILIZATION IS ',F12.2)
0231 WRITE <LUPRINT,201> ITIM,RE,R,AZ,DACRS,DALNG
0232 XDIST = XDIST + DELR
0233 C
0234 TIM = TIM + 5.0
0235 ITIM = INT<TIM>/60
0236 RE = ANOD<TIM,60.0>
0237 XL = XDIST * TXL
0238 SIGX = SORT<(XL/4.30)**2 + SIGX02>
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0239 DALINC = 4.30 * SIGX
0240 CC
0241 CC***THE ALGORITHM FOR SIGY HAS BEEN CHANGED TO CONFORM TO THE
0242 CC***ALGORITHM FOR SIGY FOUND IN SUBROUTINE DFEAP AFTER CONSUL-
0243 CC***TATION WITH DR. STEPHENS...JSH
0244 CC
0245 SIGNAL2 = (SIGAP*XDIST + SIGX0)**2
0246 SIGY = SQRT(SIGNAL2 + (.0040589 * FLOAT(IDELTH) * XDIST)**2)
0247 DWCRS = 4.36 * SIGY
0248 XC = XC + DELX
0249 YC = YC + DELY
0250 R = SURT(XC * XC + YC * YC)
0251 22 AZ = 180.0 - RADDEG * ATAN2(YC,XC)
0252 C
0253 C LABEL THE CLOUD STABILIZATION POINT WITH A *
0254 C
0255 IX = INT(0.2631 * X) + IX0
0256 IY = INT(0.3545 * Y) + IY0
0257 IF(IX.LT.0 .OR. IX.GT.9999 .OR. IY.LT.0 .OR. IY.GT.9999)GO TO 77
0258 IXX = IX
0259 IYY = IY
0260 WRITE (IPL3) I,I,IX,IY
0261 CALL SYMBL(150,150,1H+,IPL3)
0262 C
0263 C LABEL THE POINT OF MAXIMUM CONCENTRATION WITH A 0
0264 C
0265 DIR = DEGRAD * (180.0 - AVDIR)
0266 CDIR = COS(DIR)
0267 SDIR = SIN(DIR)
0268 IX1 = INT(0.2631 * (X + RNGSNC * CDIR)) + IX0
0269 IY1 = INT(0.3545 * (Y + RNGSNC * SDIR)) + IY0
0270 WRITE (IPL3) -1,I,IX1,IY1
0271 CALL SYMBL(150,150,1H0,IPL3)
0272 C
0273 C DRAW THE LINE OF CLOUD MOVEMENT ALONG THE GROUND FROM
0274 C THE CLOUD STABILIZATION POINT ON
0275 C
0276 WRITE (IPL3) -1,I,IXX,IYY
0277 RANGE = 1000.0
0278 27 X = X + RANGE * CDIR
0279 Y = Y + RANGE * SDIR
0280 IX = INT(0.2631 * X) + IX0
0281 IY = INT(0.3545 * Y) + IY0
0282 IF(IX.LT.0 .OR. IX.GT.9999 .OR. IY.LT.0 .OR. IY.GT.9999)GO TO 29
0283 WRITE (IPL3) I,I,IX,IY
0284 GO TO 27
0285 29 WRITE (IPL3) -1,I,IXX,IYY
0286 C
0287 C ARE DEFAULT CONCENTRATION VALUES GOING TO BE USED
0288 C FOR THE PLOTS
0289 C
0290 CC IF(YPOS.LT.48.0) YPOS = 458.
0291 IF(YPOS.LT.48.0 .AND. LYVERSA) YPOS = 458.
0292 CALL DREAD(NAMEF,2,ILINE)
0293 CALL LERS(YPOS,LYVERSH)
0294 IF(LYVERSH) CALL LERS(YPOS,LYVERSH)
0295 CALL CHAR(0.,YPOS,0,ILINE,64,0,0)
0296 YPOS = YPOS - 16.
0297 CALL DREAD(NAMEF,3,ILINE)
0298 LEI '05,.....5H).
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```

0299 CALL CHAR(0.,YPOS,0,ILINE,64,0,0)
0300 YPOS = YPOS - 32.
0301 C
0302 C
0303 C
0304 C
0305 C
0306 C
0307 C
0308 C
0309 C
0310 C
0311 C
0312 C
0313 C
0314 C
0315 C
0316 C
0317 C
0318 C
0319 C
0320 C
0321 C
0322 C
0323 C
0324 C
0325 C
0326 C
0327 C
0328 C
0329 C
0330 C
0331 C
0332 C
0333 C
0334 C
0335 C
0336 C
0337 C
0338 C
0339 C
0340 C
0341 C
0342 C
0343 C
0344 C
0345 C
0346 C
0347 C
0348 C
0349 C
0350 C
0351 C
0352 C
0353 C
0354 C
0355 C
0356 C
0357 C
0358 C

      YES -- SET UP THE DEFAULT VALUES
      CONC(1) = 0.1 * COMMAX
      CONC(2) = 0.5 * COMMAX
      CONC(3) = 0.75 * COMMAX
      CONC(4) = 1.0
      CALL CODE
      WRITE (IDATAF,206) CONC(1)
      CALL CHAR(40.,YPOS+40.,0,IDATAF,5,0,0)
      CALL CODE
      WRITE (IDATAF,206) CONC(2)
      CALL CHAR(120.,YPOS+35.,0,IDATAF,5,0,0)
      CALL CODE
      WRITE (IDATAF,206) CONC(3)
      CALL CHAR(256.,YPOS+32.,0,IDATAF,5,0,0)
      CALL DREAD(NAMEF,4,ILINE)
      CALL LER3(YPOS,LVERSH)
      CALL CHAR(0.,YPOS,0,ILINE,46,3,0)
      CALL CHAR(384.,YPOS,0,ILINE(25),0,3,0)
      CALL CHAR(464.,YPOS,0,ILINE(30),6,0,0)
      IF(LVERSH) GO TO 33
      READ(LU,199)IANS
      IX = 25
      IF(IANS.NE.100) GO TO 34
      IX = 30
      GO TO 34
      CONTINUE
      CALL INK(1,JTYPE,0.,0.,0,0,0,31,0,31,IX,IX)
      CONTINUE
      CALL CHAR(0.,YPOS,0,ILINE,64,0,0)
      IF(IX.LE.25)CALL CHAR(464.,YPOS,0,IERS,6,0,0)
      IF(IX.GT.25)CALL CHAR(384.,YPOS,0,IERS,0,0,0)
      YPOS = YPOS - 32.
      IF(YPOS.LT.64.0)YPOS = 458.0
      IF(YPOS.LT.64.0.AND.LVERSH)YPOS = 458.0
      DFALTC = .FALSE.
      IF(IX.LT.20)DFALTC = .TRUE.

      DO LOOP OVER THE 10 POSSIBLE CONCENTRATION VALUES FOR THE PLOTS
      IF(DFALTC)GO TO 35
      CALL DREAD(NAMEF,5,ILINE)
      CALL LER3(YPOS,LVERSH)
      CALL CHAR(0.,YPOS,0,ILINE,64,0,0)
      YPOS = YPOS - 32.
      IF(YPOS.LE.64) YPOS=458.
      35 DO 53 I=1,10
      IF DEFAULT CONCENTRATION VALUES ARE NOT BEING USED,
      READ IN THE VALUE FOR THIS PLOT
      IF(DFALTC)GO TO 37
      CALL DREAD(NAMEF,6,ILINE)
      CALL LER3(YPOS,LVERSH)
      CALL CHAR(0.,YPOS,0,ILINE,7,3,0)
      CALL CODE
      WRITE (IPX,207) I

```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0359 CALL CHAR(111, YPOS, 0, IDX, 1, 3, 0)
0360 MIN = 9
0361 CALL BLANK(IGATAF, 10)
0362 CALL INC(0, JTYPE, 144, YPOS, 0, IGATAF, MIN, 0, 31, 0, 31, IX, IY)
0363 CALL CODE
0364 READ(IGATAF, *) CONC(I)
0365 CALL CHAR(0, YPOS, 0, ILINE, 17, 0, 0)
0366 CALL CHAR(111, YPOS, 0, IDX, 1, 0, 0)
0367 YPOS = YPOS - 16
0368 IF(YPOS .LT. 40) YPOS = 458.0
0369 IF(YPOS .LT. 40) .AND. LVERS) YPOS = 458.0
0370 IF(CONC(I) .LT. 0.0) GO TO 61
0371 C
0372 C ITERATE TO FIND THE LOCATION OF THIS CONCENTRATION
0373 C ON THE PLOT
0374 C
0375 C XDIST = 0.0
0376 C DINC = 1000.0
0377 C
0378 41 CALL DFEXP(LAYTOP, CONC(I))
0379 IF(YDIST .GT. 0.0) GO TO 42
0380 XDIST = XDIST + DINC
0381 GO TO 41
0382 C
0383 42 IF(DINC .LE. 100.0) GO TO 43
0384 XDIST = XDIST - 900.0
0385 DINC = 100.0
0386 GO TO 41
0387 C
0388 43 IF(DINC .LE. 10.0) GO TO 44
0389 XDIST = XDIST - 90.0
0390 DINC = 10.0
0391 GO TO 41
0392 C
0393 C PLOT OUT THE CONCENTRATION LINE ON BOTH SIDES
0394 C
0395 XDIST = XDIST - 10.0
0396 IX = INT(0.2631 * XDIST + CDIR) + IX
0397 IY = INT(0.3545 * XDIST + SDIR) + IY
0398 IF(IX.LT.0 .OR. IX.GT.9999 .OR. IY.LT.0 .OR. IY.GT.9999) GO TO 59
0399 NUMB = 1
0400 IX(NUMB) = IX
0401 IY(NUMB) = IY
0402 NUMB = 1
0403 IX(NUMB) = IX
0404 IY(NUMB) = IY
0405 C
0406 XDIST = XDIST + 10.0
0407 IX = INT(0.2631 * (XDIST + CDIR - YDIST + SDIR)) + IX
0408 IY = INT(0.3545 * (XDIST + SDIR + YDIST + CDIR)) + IY
0409 IF(IX.LT.0 .OR. IX.GT.9999 .OR. IY.LT.0 .OR. IY.GT.9999) GO TO 59
0410 NUMB = 2
0411 IX(NUMB) = IX
0412 IY(NUMB) = IY
0413 C
0414 IX = INT(0.2631 * (XDIST + CDIR + YDIST + SDIR)) + IX
0415 IY = INT(0.3545 * (XDIST + SDIR - YDIST + CDIR)) + IY
0416 IF(IX.LT.0 .OR. IX.GT.9999 .OR. IY.LT.0 .OR. IY.GT.9999) GO TO 54
0417 NUMB = 2
0418 NUM IX
```



ORIGINAL PAGE IS  
OF POOR QUALITY

```
0419 C IYB(NUMB) = IY
0420 C
0421 C 46 XDIST = XDIST + 500.0
0422 C CALL GEXP(LAYTOP,CONC(I))
0423 C IX = INT(0.2631 * (XDIST * CDIR - YDIST * SDIR)) + IXX
0424 C IY = INT(0.3545 * (XDIST * SDIR + YDIST * CDIR)) + IYY
0425 C IF<IX,LT.0 .OR. IX.GT.9999 .OR. IY,LT.0 .OR. IY.GT.9999)GO TO 54
0426 C NUMA = NUMA + 1
0427 C IX(NUMA) = IX
0428 C IY(NUMA) = IY
0429 C
0430 C IF<YDIST .GT. 0.0)GO TO 52
0431 C NUMB = NUMB + 1
0432 C IXB(NUMB) = IX
0433 C IYB(NUMB) = IY
0434 C GO TO 54
0435 C
0436 C 52 IX = INT(0.2631 * (XDIST * CDIR + YDIST * SDIR)) + IXX
0437 C IY = INT(0.3545 * (XDIST * SDIR - YDIST * CDIR)) + IYY
0438 C IF<IX,LT.0 .OR. IX.GT.9999 .OR. IY,LT.0 .OR. IY.GT.9999)GO TO 54
0439 C NUMB = NUMB + 1
0440 C IXB(NUMB) = IX
0441 C IYB(NUMB) = IY
0442 C GO TO 46
0443 C
0444 C 54 WRITE (IPL3) -1,1,IXA(I),IYA(I)
0445 C DO 56 J=2,NUMA
0446 C WRITE (IPL3) 1,1,IXA(J),IYA(J)
0447 C IF<NUMB .EQ. 1)GO TO 59
0448 C WRITE (IPL3) -1,1,IXB(1),IYB(1)
0449 C DO 57 J=2,NUMB
0450 C WRITE (IPL3) 1,1,IXB(J),IYB(J)
0451 C 59 CONTINUE
0452 C
0453 C ON THE PLOT, CROSS OUT EITHER THE WORD FORECAST OR SOUNDING
0454 C
0455 C 61 IF<ISNDFO .NE. 0)GO TO 62
0456 C WRITE (IPL3) -1,1,707,604
0457 C WRITE (IPL3) 1,1,1174,604
0458 C GO TO 64
0459 C
0460 C 62 WRITE (IPL3) -1,1,1269,604
0461 C WRITE (IPL3) 1,1,1760,604
0462 C
0463 C PRINT OUT THE PREDICTION TIME ON THE PLOT
0464 C
0465 C 64 WRITE (IPL3) -1,1,1869,319
0466 C WRITE (IPL3,202) 100,0,0,150,ITIME,LAUNTD(4),IDAY,IMONTH,IYEAR
0467 C
0468 C IF THE LAUNCH TIME WAS ENTERED, PRINT IT OUT ON THE PLOT
0469 C
0470 C IF<.NOT. LTIME)GO TO 67
0471 C WRITE (IPL3) -1,1,1869,112
0472 C WRITE (IPL3,208) 100,0,0,150,LTIM,LAUNTD(3),LAUNTD(4),LDAY,
0473 C LMONTH,LYEAR
0474 C
0475 C ON THE PLOT, PRINT OUT THE CHARACTERS + MID 2 FOR THE LEGEND
0476 C
0477 C 67 WRITE (IPL3) -1,1,1041,1342
0478 C
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0479 WRITE (IPL3,203) 150,0,0,150,1H*
0480 WRITE (IPL3) -1,1,1041,1104
0481 WRITE (IPL3,203) 150,0,0,150,11H0
0482 C
0483 C FOR THE LEGEND ON THE PLOT, PRINT OUT THE CONCENTRATION VALUES
0484 C FOR WHICH CONTOURS WERE DRAWN
0485 C
0486 WRITE (IPL3) -1,1,1066,9587
0487 GO 75 I=1,10
0488 IF(CONC(I) .LT. 0.6)GO TO 77
0489 IF(I .NE. 1)GO TO 72
0490 WRITE (IPL3,204) 125,0,0,150,CONC(I)
0491 GO TO 73
0492
0493 72 WRITE (IPL3,205) 125,0,0,150,CONC(I)
0494 75 CONTINUE
0495 C
0496 C WRITE OUT COMMON DISK FILE
0497 C
0498 77 CALL AUDIS(NAME,1)
0499 C
0500 C CALL NGRAF TO RETURN SCOPE TO NORMAL MODE OF OPERATION
0501 C (APPROPRIATE ONLY WHEN USING PLASMASCOPE)
0502 C
0503 C CALL NGRAF
0504 C RETURN TO THE MAIN PROGRAM NEEDS
0505 C
0506 C STOP
0507 C
0508 C END OF RISOS
0509 C
0510 C END
0511 C
```

ORIGINAL PAGE IS  
OF POOR QUALITY

AR150T T=00003 IS ON CR00003 USING 00062 DLKS R=0000

```
0001 FTIM,L
0002 SUBROUTINE RWDIS(NAME, JJ)
0003 C -----
0004 C -----
0005 C -----
0006 C - THIS SUBROUTINE READS AND WRITES COMMON AREAS TO DISC
0007 C - TO BE UTILIZED BY THE VARIOUS REED PROGRAMS.
0008 C -----
0009 C -----
0010 C >>> COMMON AREAS
0011 C
0012 C
0013 C
0014 C >>> MATH PARAMETERS
0015 C COMMON P1, P10YR2, P143, IMOPI, SQR2PI, DEGRAD, RADNEG, IV2
0016 C
0017 C >>> VEHICLE PARAMETERS
0018 C COMMON VPAK(16), SIGHCL
0019 C
0020 C >>> OPTION PARAMETERS
0021 C COMMON YPOS, YES, ISMFO, IVHICL, IRUN, ICALC, IPLACE, ITIMEZ, IGTMP,
0022 C NUMRUM, LUD, NUM, MONLY, NFORM, LUPRNT, ITOU, IPL1, IPL2, IPL3
0023 C INTEGER YES
0024 C
0025 C >>> NET DATA
0026 C COMMON ALT(30), IDIR(30), SPEED(30), TEMP(30), PRESS(30), PTEMP(30),
0027 C SURDEN, FILE(3)
0028 C INTEGER FILE
0029 C
0030 C >>> CALCULATION TIME
0031 C COMMON ITIME, IDAY, IMONTH(2), IYEAR
0032 C
0033 C >>> SOUNDING TIME
0034 C COMMON ISTIME, ISDAY, ISHOK(2), ISYEAR
0035 C
0036 C >>> LAUNCH TIME
0037 C COMMON LTIME, LTIM, LDAY, LMONTH(2), LYEAR, LAUNTC(10)
0038 C LOGICAL LTIME
0039 C
0040 C >>> CLOUD STABILIZATION AND LAYER PARAMETERS
0041 C COMMON STBLT, STBTZ, STBRNG, STBTIM, LLDGRAD, RISTIN(30), BOTLAY,
0042 C TOPLAY, CALHT, LAYTOP, LAYBOT, REFLEC, IPTZ
0043 C
0044 C >>> DIFFUSION COEFFICIENTS
0045 C COMMON SIGX, SIGY, SIGZ, SIGAP, XDIST, YDIST, EXPZ, SASIGZ, SIGY, SIGAL,
0046 C CLDLNG, AVSPD, AVDIR, SIGAZ
0047 C
0048 C >>> CONCENTRATION AND DOSAGE VALUES AND RANGES
0049 C COMMON CONMAX, CONCPK, DOSMAX, DOSPK, RMSMNC, RMSMD
0050 C
0051 C >>> EQUIVALENCE STATEMENT FOR VEHICLE PARAMETERS
0052 C EQUIVALENCE (Q1, VPAK(1)), (Q2, VPAK(2)), (Q3, VPAK(3)),
0053 C (Q4, VPAK(4)), (Q5, VPAK(5)), (Q6, VPAK(6)),
0054 C (Q7, VPAK(7)), (Q8, VPAK(8)), (Q9, VPAK(9)),
0055 C (Q10, VPAK(10)), (Q11, VPAK(11)), (Q12, VPAK(12)),
0056 C (Q13, VPAK(13)), (Q14, VPAK(14)), (Q15, VPAK(15)),
0057 C (Q16, VPAK(16)), (Q17, VPAK(17)), (Q18, VPAK(18)),
0058 C (Q19, VPAK(19)), (Q20, VPAK(20)), (Q21, VPAK(21)), (Q22, VPAK(22)),
0059 C (Q23, VPAK(23)), (Q24, VPAK(24)), (Q25, VPAK(25)), (Q26, VPAK(26)),
0060 C (Q27, VPAK(27)), (Q28, VPAK(28)), (Q29, VPAK(29)), (Q30, VPAK(30)),
0061 C (Q31, VPAK(31)), (Q32, VPAK(32)), (Q33, VPAK(33)), (Q34, VPAK(34)),
0062 C (Q35, VPAK(35)), (Q36, VPAK(36)), (Q37, VPAK(37)), (Q38, VPAK(38)),
0063 C (Q39, VPAK(39)), (Q40, VPAK(40)), (Q41, VPAK(41)), (Q42, VPAK(42)),
0064 C (Q43, VPAK(43)), (Q44, VPAK(44)), (Q45, VPAK(45)), (Q46, VPAK(46)),
0065 C (Q47, VPAK(47)), (Q48, VPAK(48)), (Q49, VPAK(49)), (Q50, VPAK(50)),
0066 C (Q51, VPAK(51)), (Q52, VPAK(52)), (Q53, VPAK(53)), (Q54, VPAK(54)),
0067 C (Q55, VPAK(55)), (Q56, VPAK(56)), (Q57, VPAK(57)), (Q58, VPAK(58)),
0068 C (Q59, VPAK(59)), (Q60, VPAK(60)), (Q61, VPAK(61)), (Q62, VPAK(62)),
0069 C (Q63, VPAK(63)), (Q64, VPAK(64)), (Q65, VPAK(65)), (Q66, VPAK(66)),
0070 C (Q67, VPAK(67)), (Q68, VPAK(68)), (Q69, VPAK(69)), (Q70, VPAK(70)),
0071 C (Q71, VPAK(71)), (Q72, VPAK(72)), (Q73, VPAK(73)), (Q74, VPAK(74)),
0072 C (Q75, VPAK(75)), (Q76, VPAK(76)), (Q77, VPAK(77)), (Q78, VPAK(78)),
0073 C (Q79, VPAK(79)), (Q80, VPAK(80)), (Q81, VPAK(81)), (Q82, VPAK(82)),
0074 C (Q83, VPAK(83)), (Q84, VPAK(84)), (Q85, VPAK(85)), (Q86, VPAK(86)),
0075 C (Q87, VPAK(87)), (Q88, VPAK(88)), (Q89, VPAK(89)), (Q90, VPAK(90)),
0076 C (Q91, VPAK(91)), (Q92, VPAK(92)), (Q93, VPAK(93)), (Q94, VPAK(94)),
0077 C (Q95, VPAK(95)), (Q96, VPAK(96)), (Q97, VPAK(97)), (Q98, VPAK(98)),
0078 C (Q99, VPAK(99)), (Q100, VPAK(100)), (Q101, VPAK(101)), (Q102, VPAK(102)),
0079 C (Q103, VPAK(103)), (Q104, VPAK(104)), (Q105, VPAK(105)), (Q106, VPAK(106)),
0080 C (Q107, VPAK(107)), (Q108, VPAK(108)), (Q109, VPAK(109)), (Q110, VPAK(110)),
0081 C (Q111, VPAK(111)), (Q112, VPAK(112)), (Q113, VPAK(113)), (Q114, VPAK(114)),
0082 C (Q115, VPAK(115)), (Q116, VPAK(116)), (Q117, VPAK(117)), (Q118, VPAK(118)),
0083 C (Q119, VPAK(119)), (Q120, VPAK(120)), (Q121, VPAK(121)), (Q122, VPAK(122)),
0084 C (Q123, VPAK(123)), (Q124, VPAK(124)), (Q125, VPAK(125)), (Q126, VPAK(126)),
0085 C (Q127, VPAK(127)), (Q128, VPAK(128)), (Q129, VPAK(129)), (Q130, VPAK(130)),
0086 C (Q131, VPAK(131)), (Q132, VPAK(132)), (Q133, VPAK(133)), (Q134, VPAK(134)),
0087 C (Q135, VPAK(135)), (Q136, VPAK(136)), (Q137, VPAK(137)), (Q138, VPAK(138)),
0088 C (Q139, VPAK(139)), (Q140, VPAK(140)), (Q141, VPAK(141)), (Q142, VPAK(142)),
0089 C (Q143, VPAK(143)), (Q144, VPAK(144)), (Q145, VPAK(145)), (Q146, VPAK(146)),
0090 C (Q147, VPAK(147)), (Q148, VPAK(148)), (Q149, VPAK(149)), (Q150, VPAK(150)),
0091 C (Q151, VPAK(151)), (Q152, VPAK(152)), (Q153, VPAK(153)), (Q154, VPAK(154)),
0092 C (Q155, VPAK(155)), (Q156, VPAK(156)), (Q157, VPAK(157)), (Q158, VPAK(158)),
0093 C (Q159, VPAK(159)), (Q160, VPAK(160)), (Q161, VPAK(161)), (Q162, VPAK(162)),
0094 C (Q163, VPAK(163)), (Q164, VPAK(164)), (Q165, VPAK(165)), (Q166, VPAK(166)),
0095 C (Q167, VPAK(167)), (Q168, VPAK(168)), (Q169, VPAK(169)), (Q170, VPAK(170)),
0096 C (Q171, VPAK(171)), (Q172, VPAK(172)), (Q173, VPAK(173)), (Q174, VPAK(174)),
0097 C (Q175, VPAK(175)), (Q176, VPAK(176)), (Q177, VPAK(177)), (Q178, VPAK(178)),
0098 C (Q179, VPAK(179)), (Q180, VPAK(180)), (Q181, VPAK(181)), (Q182, VPAK(182)),
0099 C (Q183, VPAK(183)), (Q184, VPAK(184)), (Q185, VPAK(185)), (Q186, VPAK(186)),
0100 C (Q187, VPAK(187)), (Q188, VPAK(188)), (Q189, VPAK(189)), (Q190, VPAK(190)),
0101 C (Q191, VPAK(191)), (Q192, VPAK(192)), (Q193, VPAK(193)), (Q194, VPAK(194)),
0102 C (Q195, VPAK(195)), (Q196, VPAK(196)), (Q197, VPAK(197)), (Q198, VPAK(198)),
0103 C (Q199, VPAK(199)), (Q200, VPAK(200)), (Q201, VPAK(201)), (Q202, VPAK(202)),
0104 C (Q203, VPAK(203)), (Q204, VPAK(204)), (Q205, VPAK(205)), (Q206, VPAK(206)),
0105 C (Q207, VPAK(207)), (Q208, VPAK(208)), (Q209, VPAK(209)), (Q210, VPAK(210)),
0106 C (Q211, VPAK(211)), (Q212, VPAK(212)), (Q213, VPAK(213)), (Q214, VPAK(214)),
0107 C (Q215, VPAK(215)), (Q216, VPAK(216)), (Q217, VPAK(217)), (Q218, VPAK(218)),
0108 C (Q219, VPAK(219)), (Q220, VPAK(220)), (Q221, VPAK(221)), (Q222, VPAK(222)),
0109 C (Q223, VPAK(223)), (Q224, VPAK(224)), (Q225, VPAK(225)), (Q226, VPAK(226)),
0110 C (Q227, VPAK(227)), (Q228, VPAK(228)), (Q229, VPAK(229)), (Q230, VPAK(230)),
0111 C (Q231, VPAK(231)), (Q232, VPAK(232)), (Q233, VPAK(233)), (Q234, VPAK(234)),
0112 C (Q235, VPAK(235)), (Q236, VPAK(236)), (Q237, VPAK(237)), (Q238, VPAK(238)),
0113 C (Q239, VPAK(239)), (Q240, VPAK(240)), (Q241, VPAK(241)), (Q242, VPAK(242)),
0114 C (Q243, VPAK(243)), (Q244, VPAK(244)), (Q245, VPAK(245)), (Q246, VPAK(246)),
0115 C (Q247, VPAK(247)), (Q248, VPAK(248)), (Q249, VPAK(249)), (Q250, VPAK(250)),
0116 C (Q251, VPAK(251)), (Q252, VPAK(252)), (Q253, VPAK(253)), (Q254, VPAK(254)),
0117 C (Q255, VPAK(255)), (Q256, VPAK(256)), (Q257, VPAK(257)), (Q258, VPAK(258)),
0118 C (Q259, VPAK(259)), (Q260, VPAK(260)), (Q261, VPAK(261)), (Q262, VPAK(262)),
0119 C (Q263, VPAK(263)), (Q264, VPAK(264)), (Q265, VPAK(265)), (Q266, VPAK(266)),
0120 C (Q267, VPAK(267)), (Q268, VPAK(268)), (Q269, VPAK(269)), (Q270, VPAK(270)),
0121 C (Q271, VPAK(271)), (Q272, VPAK(272)), (Q273, VPAK(273)), (Q274, VPAK(274)),
0122 C (Q275, VPAK(275)), (Q276, VPAK(276)), (Q277, VPAK(277)), (Q278, VPAK(278)),
0123 C (Q279, VPAK(279)), (Q280, VPAK(280)), (Q281, VPAK(281)), (Q282, VPAK(282)),
0124 C (Q283, VPAK(283)), (Q284, VPAK(284)), (Q285, VPAK(285)), (Q286, VPAK(286)),
0125 C (Q287, VPAK(287)), (Q288, VPAK(288)), (Q289, VPAK(289)), (Q290, VPAK(290)),
0126 C (Q291, VPAK(291)), (Q292, VPAK(292)), (Q293, VPAK(293)), (Q294, VPAK(294)),
0127 C (Q295, VPAK(295)), (Q296, VPAK(296)), (Q297, VPAK(297)), (Q298, VPAK(298)),
0128 C (Q299, VPAK(299)), (Q300, VPAK(300)), (Q301, VPAK(301)), (Q302, VPAK(302)),
0129 C (Q303, VPAK(303)), (Q304, VPAK(304)), (Q305, VPAK(305)), (Q306, VPAK(306)),
0130 C (Q307, VPAK(307)), (Q308, VPAK(308)), (Q309, VPAK(309)), (Q310, VPAK(310)),
0131 C (Q311, VPAK(311)), (Q312, VPAK(312)), (Q313, VPAK(313)), (Q314, VPAK(314)),
0132 C (Q315, VPAK(315)), (Q316, VPAK(316)), (Q317, VPAK(317)), (Q318, VPAK(318)),
0133 C (Q319, VPAK(319)), (Q320, VPAK(320)), (Q321, VPAK(321)), (Q322, VPAK(322)),
0134 C (Q323, VPAK(323)), (Q324, VPAK(324)), (Q325, VPAK(325)), (Q326, VPAK(326)),
0135 C (Q327, VPAK(327)), (Q328, VPAK(328)), (Q329, VPAK(329)), (Q330, VPAK(330)),
0136 C (Q331, VPAK(331)), (Q332, VPAK(332)), (Q333, VPAK(333)), (Q334, VPAK(334)),
0137 C (Q335, VPAK(335)), (Q336, VPAK(336)), (Q337, VPAK(337)), (Q338, VPAK(338)),
0138 C (Q339, VPAK(339)), (Q340, VPAK(340)), (Q341, VPAK(341)), (Q342, VPAK(342)),
0139 C (Q343, VPAK(343)), (Q344, VPAK(344)), (Q345, VPAK(345)), (Q346, VPAK(346)),
0140 C (Q347, VPAK(347)), (Q348, VPAK(348)), (Q349, VPAK(349)), (Q350, VPAK(350)),
0141 C (Q351, VPAK(351)), (Q352, VPAK(352)), (Q353, VPAK(353)), (Q354, VPAK(354)),
0142 C (Q355, VPAK(355)), (Q356, VPAK(356)), (Q357, VPAK(357)), (Q358, VPAK(358)),
0143 C (Q359, VPAK(359)), (Q360, VPAK(360)), (Q361, VPAK(361)), (Q362, VPAK(362)),
0144 C (Q363, VPAK(363)), (Q364, VPAK(364)), (Q365, VPAK(365)), (Q366, VPAK(366)),
0145 C (Q367, VPAK(367)), (Q368, VPAK(368)), (Q369, VPAK(369)), (Q370, VPAK(370)),
0146 C (Q371, VPAK(371)), (Q372, VPAK(372)), (Q373, VPAK(373)), (Q374, VPAK(374)),
0147 C (Q375, VPAK(375)), (Q376, VPAK(376)), (Q377, VPAK(377)), (Q378, VPAK(378)),
0148 C (Q379, VPAK(379)), (Q380, VPAK(380)), (Q381, VPAK(381)), (Q382, VPAK(382)),
0149 C (Q383, VPAK(383)), (Q384, VPAK(384)), (Q385, VPAK(385)), (Q386, VPAK(386)),
0150 C (Q387, VPAK(387)), (Q388, VPAK(388)), (Q389, VPAK(389)), (Q390, VPAK(390)),
0151 C (Q391, VPAK(391)), (Q392, VPAK(392)), (Q393, VPAK(393)), (Q394, VPAK(394)),
0152 C (Q395, VPAK(395)), (Q396, VPAK(396)), (Q397, VPAK(397)), (Q398, VPAK(398)),
0153 C (Q399, VPAK(399)), (Q400, VPAK(400)), (Q401, VPAK(401)), (Q402, VPAK(402)),
0154 C (Q403, VPAK(403)), (Q404, VPAK(404)), (Q405, VPAK(405)), (Q406, VPAK(406)),
0155 C (Q407, VPAK(407)), (Q408, VPAK(408)), (Q409, VPAK(409)), (Q410, VPAK(410)),
0156 C (Q411, VPAK(411)), (Q412, VPAK(412)), (Q413, VPAK(413)), (Q414, VPAK(414)),
0157 C (Q415, VPAK(415)), (Q416, VPAK(416)), (Q417, VPAK(417)), (Q418, VPAK(418)),
0158 C (Q419, VPAK(419)), (Q420, VPAK(420)), (Q421, VPAK(421)), (Q422, VPAK(422)),
0159 C (Q423, VPAK(423)), (Q424, VPAK(424)), (Q425, VPAK(425)), (Q426, VPAK(426)),
0160 C (Q427, VPAK(427)), (Q428, VPAK(428)), (Q429, VPAK(429)), (Q430, VPAK(430)),
0161 C (Q431, VPAK(431)), (Q432, VPAK(432)), (Q433, VPAK(433)), (Q434, VPAK(434)),
0162 C (Q435, VPAK(435)), (Q436, VPAK(436)), (Q437, VPAK(437)), (Q438, VPAK(438)),
0163 C (Q439, VPAK(439)), (Q440, VPAK(440)), (Q441, VPAK(441)), (Q442, VPAK(442)),
0164 C (Q443, VPAK(443)), (Q444, VPAK(444)), (Q445, VPAK(445)), (Q446, VPAK(446)),
0165 C (Q447, VPAK(447)), (Q448, VPAK(448)), (Q449, VPAK(449)), (Q450, VPAK(450)),
0166 C (Q451, VPAK(451)), (Q452, VPAK(452)), (Q453, VPAK(453)), (Q454, VPAK(454)),
0167 C (Q455, VPAK(455)), (Q456, VPAK(456)), (Q457, VPAK(457)), (Q458, VPAK(458)),
0168 C (Q459, VPAK(459)), (Q460, VPAK(460)), (Q461, VPAK(461)), (Q462, VPAK(462)),
0169 C (Q463, VPAK(463)), (Q464, VPAK(464)), (Q465, VPAK(465)), (Q466, VPAK(466)),
0170 C (Q467, VPAK(467)), (Q468, VPAK(468)), (Q469, VPAK(469)), (Q470, VPAK(470)),
0171 C (Q471, VPAK(471)), (Q472, VPAK(472)), (Q473, VPAK(473)), (Q474, VPAK(474)),
0172 C (Q475, VPAK(475)), (Q476, VPAK(476)), (Q477, VPAK(477)), (Q478, VPAK(478)),
0173 C (Q479, VPAK(479)), (Q480, VPAK(480)), (Q481, VPAK(481)), (Q482, VPAK(482)),
0174 C (Q483, VPAK(483)), (Q484, VPAK(484)), (Q485, VPAK(485)), (Q486, VPAK(486)),
0175 C (Q487, VPAK(487)), (Q488, VPAK(488)), (Q489, VPAK(489)), (Q490, VPAK(490)),
0176 C (Q491, VPAK(491)), (Q492, VPAK(492)), (Q493, VPAK(493)), (Q494, VPAK(494)),
0177 C (Q495, VPAK(495)), (Q496, VPAK(496)), (Q497, VPAK(497)), (Q498, VPAK(498)),
0178 C (Q499, VPAK(499)), (Q500, VPAK(500)), (Q501, VPAK(501)), (Q502, VPAK(502)),
0179 C (Q503, VPAK(503)), (Q504, VPAK(504)), (Q505, VPAK(505)), (Q506, VPAK(506)),
0180 C (Q507, VPAK(507)), (Q508, VPAK(508)), (Q509, VPAK(509)), (Q510, VPAK(510)),
0181 C (Q511, VPAK(511)), (Q512, VPAK(512)), (Q513, VPAK(513)), (Q514, VPAK(514)),
0182 C (Q515, VPAK(515)), (Q516, VPAK(516)), (Q517, VPAK(517)), (Q518, VPAK(518)),
0183 C (Q519, VPAK(519)), (Q520, VPAK(520)), (Q521, VPAK(521)), (Q522, VPAK(522)),
0184 C (Q523, VPAK(523)), (Q524, VPAK(524)), (Q525, VPAK(525)), (Q526, VPAK(526)),
0185 C (Q527, VPAK(527)), (Q528, VPAK(528)), (Q529, VPAK(529)), (Q530, VPAK(530)),
0186 C (Q531, VPAK(531)), (Q532, VPAK(532)), (Q533, VPAK(533)), (Q534, VPAK(534)),
0187 C (Q535, VPAK(535)), (Q536, VPAK(536)), (Q537, VPAK(537)), (Q538, VPAK(538)),
0188 C (Q539, VPAK(539)), (Q540, VPAK(540)), (Q541, VPAK(541)), (Q542, VPAK(542)),
0189 C (Q543, VPAK(543)), (Q544, VPAK(544)), (Q545, VPAK(545)), (Q546, VPAK(546)),
0190 C (Q547, VPAK(547)), (Q548, VPAK(548)), (Q549, VPAK(549)), (Q550, VPAK(550)),
0191 C (Q551, VPAK(551)), (Q552, VPAK(552)), (Q553, VPAK(553)), (Q554, VPAK(554)),
0192 C (Q555, VPAK(555)), (Q556, VPAK(556)), (Q557, VPAK(557)), (Q558, VPAK(558)),
0193 C (Q559, VPAK(559)), (Q560, VPAK(560)), (Q561, VPAK(561)), (Q562, VPAK(562)),
0194 C (Q563, VPAK(563)), (Q564, VPAK(564)), (Q565, VPAK(565)), (Q566, VPAK(566)),
0195 C (Q567, VPAK(567)), (Q568, VPAK(568)), (Q569, VPAK(569)), (Q570, VPAK(570)),
0196 C (Q571, VPAK(571)), (Q572, VPAK(572)), (Q573, VPAK(573)), (Q574, VPAK(574)),
0197 C (Q575, VPAK(575)), (Q576, VPAK(576)), (Q577, VPAK(577)), (Q578, VPAK(578)),
0198 C (Q579, VPAK(579)), (Q580, VPAK(580)), (Q581, VPAK(581)), (Q582, VPAK(582)),
0199 C (Q583, VPAK(583)), (Q584, VPAK(584)), (Q585, VPAK(585)), (Q586, VPAK(586)),
0200 C (Q587, VPAK(587)), (Q588, VPAK(588)), (Q589, VPAK(589)), (Q590, VPAK(590)),
0201 C (Q591, VPAK(591)), (Q592, VPAK(592)), (Q593, VPAK(593)), (Q594, VPAK(594)),
0202 C (Q595, VPAK(595)), (Q596, VPAK(596)), (Q597, VPAK(597)), (Q598, VPAK(598)),
0203 C (Q599, VPAK(599)), (Q600, VPAK(600)), (Q601, VPAK(601)), (Q602, VPAK(602)),
0204 C (Q603, VPAK(603)), (Q604, VPAK(604)), (Q605, VPAK(605)), (Q606, VPAK(606)),
0205 C (Q607, VPAK(607)), (Q608, VPAK(608)), (Q609, VPAK(609)), (Q610, VPAK(610)),
0206 C (Q611, VPAK(611)), (Q612, VPAK(612)), (Q613, VPAK(613)), (Q614, VPAK(614)),
0207 C (Q615, VPAK(615)), (Q616, VPAK(616)), (Q617, VPAK(617)), (Q618, VPAK(618)),
0208 C (Q619, VPAK(619)), (Q620, VPAK(620)), (Q621, VPAK(621)), (Q622, VPAK(622)),
0209 C (Q623, VPAK(623)), (Q624, VPAK(624)), (Q625, VPAK(625)), (Q626, VPAK(626)),
0210 C (Q627, VPAK(627)), (Q628, VPAK(628)), (Q629, VPAK(629)), (Q630, VPAK(630)),
0211 C (Q631, VPAK(631)), (Q632, VPAK(632)), (Q633, VPAK(633)), (Q634, VPAK(634)),
0212 C (Q635, VPAK(635)), (Q636, VPAK(636)), (Q637, VPAK(637)), (Q638, VPAK(638)),
0213 C (Q639, VPAK(639)), (Q640, VPAK(640)), (Q641, VPAK(641)), (Q642, VPAK(642)),
0214 C (Q643, VPAK(643)), (Q644, VPAK(644)), (Q645, VPAK(645)), (Q646, VPAK(646)),
0215 C (Q647, VPAK(647)), (Q648, VPAK(648)), (Q649, VPAK(649)), (Q650, VPAK(650)),
0216 C (Q651, VPAK(651)), (Q652, VPAK(652)), (Q653, VPAK(653)), (Q654, VPAK(654)),
0217 C (Q655, VPAK(655)), (Q656, VPAK(656)), (Q657, VPAK(657)), (Q658, VPAK(658)),
0218 C (Q659, VPAK(659)), (Q660, VPAK(660)), (Q661, VPAK(661)), (Q662, VPAK(662)),
0219 C (Q663, VPAK(663)), (Q664, VPAK(664)), (Q665, VPAK(665)), (Q666, VPAK(666)),
0220 C (Q667, VPAK(667)), (Q668, VPAK(668)), (Q669, VPAK(669)), (Q670, VPAK(670)),
0221 C (Q671, VPAK(671)), (Q672, VPAK(672)), (Q673, VPAK(673)), (Q674, VPAK(674)),
0222 C (Q675, VPAK(675)), (Q676, VPAK(676)), (Q677, VPAK(677)), (Q678, VPAK(678)),
0223 C (Q679, VPAK(679)), (Q680, VPAK(680)), (Q681, VPAK(681)), (Q682, VPAK(682)),
0224 C (Q683, VPAK(683)), (Q684, VPAK(684)), (Q685, VPAK(685)), (Q686, VPAK(686)),
0225 C (Q687, VPAK(687)), (Q688, VPAK(688)), (Q689, VPAK(689)), (Q690, VPAK(690)),
0226 C (Q691, VPAK(691)), (Q692, VPAK(692)), (Q693, VPAK(693)), (Q694, VPAK(694)),
0227 C (Q695, VPAK(695)), (Q696, VPAK(696)), (Q697, VPAK(697)), (Q698, VPAK(698)),
0228 C (Q699, VPAK(699)), (Q700, VPAK(700)), (Q701, VPAK(701)), (Q702, VPAK(702)),
0229 C (Q703, VPAK(703)), (Q704, VPAK(704)), (Q705, VPAK(705)), (Q706, VPAK(706)),
0230 C (Q707, VPAK(707)), (Q708, VPAK(708)), (Q709, VPAK(709)), (Q710, VPAK(710)),
0231 C (Q711, VPAK(711)), (Q712, VPAK(712)), (Q713, VPAK(713)), (Q714, VPAK(714)),
0232 C (Q715, VPAK(715)), (Q716, VPAK(716)), (Q717, VPAK(717)), (Q718, VPAK(718)),
0233 C (Q719, VPAK(719)), (Q720, VPAK(720)), (Q721, VPAK(721)), (Q722, VPAK(722)),
0234 C (Q723, VPAK(723)), (Q724, VPAK(724)), (Q725, VPAK(725)), (Q726, VPAK(726)),
0235 C (Q727, VPAK(727)), (Q728, VPAK(728)), (Q729, VPAK(729)), (Q730, VPAK(730)),
0236 C (Q731, VPAK(731)), (Q732, VPAK(732)), (Q733, VPAK(733)), (Q734, VPAK(734)),
0237 C (Q735, VPAK(735)), (Q736, VPAK(736)), (Q737, VPAK(737)), (Q738, VPAK(738)),
0238 C (Q739, VPAK(739)), (Q740, VPAK(740)), (Q741, VPAK(741)), (Q742, VPAK(742)),
0239 C (Q743, VPAK(743)), (Q744, VPAK(744)), (Q745, VPAK(745)), (Q746, VPAK(746)),
0240 C (Q747, VPAK(747)), (Q748, VPAK(748)), (Q749, VPAK(749)), (Q750, VPAK(750
```

ORIGINAL PAGE IS  
OF POOR QUALITY

```

0059 DIMENSION NAME(3)
0060 EQUIVALENCE (OBUF(1),PI)
0061 CALL GFEK(OUCB, IERR, NAME, 0)
0062 IF(IERR .LT. 0)CALL CREAT(OUCB, IERR, NAME, 1, 3, 0, 0, 144)
0063 IF(JJ.EQ.1)CALL WFIT(OUCB, IERR, OBUF, 557)
0064 IF(JJ.EQ.0)CALL RENOF(OUCB, IERR, OBUF, 557)
0065 CALL CLOSE(OUCB, IERR)
0066 RETURN
0067 END
0068 SUBROUTINE DFEXP(J, CONC)
0069
0070 C
0071 C
0072 C
0073 C
0074 C
0075 C
0076 C
0077 C
0078 C
0079 C
0080 C
0081 C
0082 C
0083 C
0084 C
0085 C
0086 C
0087 C
0088 C
0089 C
0090 C
0091 C
0092 C
0093 C
0094 C
0095 C
0096 C
0097 C
0098 C
0099 C
0100 C
0101 C
0102 C
0103 C
0104 C
0105 C
0106 C
0107 C
0108 C
0109 C
0110 C
0111 C
0112 C
0113 C
0114 C
0115 C
0116 C
0117 C
0118 C

```

THIS SUBROUTINE CALCULATES DIFFUSION EXPONENTIALS  
 J - INDEX IN THE ALT ARRAY OF THE TOP OF THE LAYER  
 CONC - CONCENTRATION TO BE TESTED  
 -----  
 >>> COMMON AREAS  
 >>> MATH PARAMETERS  
 COMMON PI, P1OVR2, P143, TQOPT, SQRP1, DEGRAD, RADDEG, IV2  
 >>> VEHICLE PARAMETERS  
 COMMON VPAR(10), SIGACL  
 >>> OPTION PARAMETERS  
 COMMON YPOS, YES, ISMOFO, IVHICL, IRUN, ICALC, IPLACE, ITIMEZ, IGOIMP,  
 HURUN, LU, NUM, MOHLY, MFORM, LUPRNT, ITDU, IPL1, IPL2, IPL3  
 INTEGER YES  
 >>> MET DATA  
 COMMON ALT(30), IDIK(30), SPEED(30), TEMP(30), PRESS(30), PTEMP(30),  
 SURDEN, FILE(3)  
 INTEGER FILE  
 >>> CALCULATION TIME  
 COMMON ITIME, IDAY, IMONTH(2), IYEAR  
 >>> SOUNDING TIME  
 COMMON ISIME, ISDAY, ISMOIK(2), ISYEAR  
 >>> LAUNCH TIME  
 COMMON LTIME, LTIM, LDAY, LMONTH(2), LYEAR, LAUNTC(10)  
 LOGICAL LTIME  
 >>> CLOUD STABILIZATION AND LAYER PARAMETERS  
 COMMON STBLT, STBHT, STBAZ, STBRNG, STBTH, CLDRAD, RSTIK(30), BOTLAY,  
 TOPLAY, CALHT, LAYTOP, LAYBOT, REFLEC, IPTZ  
 >>> DIFFUSION COEFFICIENTS  
 COMMON SIGX, SIGY, SIGZ, SIGAP, XDIST, YDIST, EXPZ, SASIGZ, SIGY, SIGAL,  
 TELNG, AVSFD, AVDIR, SIGAZ  
 >>> CONCENTRATION AND DOSAGE VALUES AND RANGES  
 COMMON CONCX, CONCY, CONSZ, DUSMAY, DUSLEV, QUESM, QUESPM

ORIGINAL PAGE IS  
OF POOR QUALITY

```

0119 C >>> EQUIVALENCE STATEMENT FOR VEHICLE PARAMETERS
0120 EQUIVALENCE (Q1,VPAR(1)),(Q12,VPAR(2)),(Q13,VPAR(3)),
0121 (Q14,VPAR(4)),(Q15,VPAR(5)),(Q16,VPAR(6)),
0122 (Q17,VPAR(7)),(Q18,VPAR(8)),(Q19,VPAR(9)),
0123 (HEATN,VPAR(10)),(HEATM,VPAR(11)),(HEATA,VPAR(12)),
0124 (PHCL,VPAR(13)),(PCO,VPAR(14)),(PCO2,VPAR(15)),
0125 (PAL203,VPAR(16)),(PHG,VPAR(17)),(GAMMAX,VPAR(18))
0126
0127 C
0128 C CALCULATE SIGMA Z
0129 C
0130 SIGZ = XDIST * SIGAP + SIGZ0/1.28
0131 SOSIGZ = 2.0 * SIGZ * SIGZ
0132 C
0133 C CALCULATE THE EXPONENTIAL SUM IN THE DIFFUSION EQUATION
0134 C
0135 TWOI = 2.0
0136 ZT = ALT(J)
0137 TEMP2 = STBHT - CALHT
0138 TEMP3 = STBHT - 2.0 * BOTLAY + CALHT
0139 EXPZ = EXP(-TEMP2/SOSIGZ) +
0140 EXP(-TEMP3 * TEMP3/SOSIGZ)
0141 4 TEMP1 = TWOI * (ZT - BOTLAY)
0142 TEXPSM = EXPZ
0143 TEMP = (TEMP1 - TEMP2)**2/SOSIGZ
0144 IF(TEMP .LE. 120.0)EXPZ = EXPZ + EXP(-TEMP)
0145 TEMP = (TEMP1 + TEMP2)**2/SOSIGZ
0146 IF(TEMP .LE. 120.0)EXPZ = EXPZ + EXP(-TEMP)
0147 TEMP = (TEMP1 - TEMP3)**2/SOSIGZ
0148 IF(TEMP .LE. 120.0)EXPZ = EXPZ + EXP(-TEMP)
0149 TEMP = (TEMP1 + TEMP3)**2/SOSIGZ
0150 IF(TEMP .LE. 120.0)EXPZ = EXPZ + EXP(-TEMP)
0151 IF(EXPZ .EQ. TEXPSM)GO TO 7
0152 TWOI = TWOI + 2.0
0153 GO TO 4
0154 7 EXPZ = REFLEC * EXPZ
0155 C
0156 C CALCULATE SIGMA Y
0157 C
0158 SIGAL = XDIST * SIGAP + SIGX0
0159 CC
0160 CC***THE FOLLOWING TWO LINES WERE ADDED TO ENSURE THAT THE DIFF-
0161 CC***ERENCE IN THE DIRECTIONS IS ALWAYS LESS THAN 180 DEGREES.
0162 CC
0163 ITHET = IABS(IDIR(J) - IDIR(I))
0164 IF(ITHET.GT.180)ITHET = 360 - ITHET
0165 SIGY = SORT(SIGAL * SIGAL +
0166 (0.0040589 * FLOAT(ITHET) * XDIST)**2)
0167 C
0168 C CALCULATE CLOUD LENGTH
0169 C
0170 DELVEL = SPEED(J) - SPEED(I)
0171 CLOUDNG = 0.28 * DELVEL * XDIST/AVSPD
0172 IF(DELVEL .GE. 0.0)GO TO 11
0173 IF(PTEMP(J)-PTEMP(I) .GT. 0.0)CLOUDNG = 0.0
0174 CLOUDNG = ABS(CLOUDNG)
0175 C
0176 C CALCULATE SIGMA X
0177 C
0178 11 SIGX = SORT((CLOUDNG/4.3)**2 + SIGX0 * SIGX0)

```



ORIGINAL PAGE IS  
OF POOR QUALITY

```
0239 COMMON LTIME, LTIM, LDAY, LMONTH(2), LYEAR, LAUNTD(10)
0240 LOGICAL LTIME
0241 C >>> CLOUD STABILIZATION AND LAYER PARAMETERS
0242 COMMON STBLT, STBLT, STRAZD, STRNGC, STBTIN, CLDRAD, RISTJK(30), BOTLAY,
0243 TOPLAY, CALNT, LAYTOP, LAYBOT, REFLEC, IPTZ
0244 C >>> DIFFUSION COEFFICIENTS
0245 COMMON SIGX0, SIGX, SIGZ0, SIGAP, XDIST, YDIST, EXPZ, SOSIGZ, SIGY, SIGAL,
0246 CLDNG, AVSFD, AVDIR, SIGAZ
0249 C >>> CONCENTRATION AND DOSAGE VALUES AND RANGES
0250 COMMON CONMAX, CONCPK, DOSMAX, DOSPK, RINGSMC, RINGSND
0251 C >>> EQUIVALENCE STATEMENT FOR VEHICLE PARAMETERS
0252 EQUIVALENCE (QC1, VPAR(1)), (QC2, VPAR(2)), (QC3, VPAR(3)),
0253 (QT1, VPAR(4)), (QT2, VPAR(5)), (QT3, VPAR(6)),
0254 (AA, VPAR(7)), (BB, VPAR(8)), (CC, VPAR(9)),
0255 (HEATN, VPAR(10)), (HEATA, VPAR(11)), (HEATA, VPAR(12)),
0256 (PHCL, VPAR(13)), (PCO, VPAR(14)), (PCO2, VPAR(15)),
0257 (PAL203, VPAR(16)), (PNO, VPAR(17)), (GAMMAX, VPAR(18))
0258 DIMENSION ILINE(32), IDATAF(10), IERS(32), IMAPL(46), NAMEF(3)
0259
0260 C INPUT FORMAT STATEMENT
0261 C
0262 C 100 FORMAT (12,1XA1)
0263 C
0264 C OUTPUT FORMAT STATEMENT
0265 C
0266 C DIMENSION STATEMENT
0267 C
0268 C DIMENSION IX(8), IY(8)
0269 C
0270 C DATA STATEMENTS
0271 C
0272 C DATA IERS/32*2H /
0273 C DATA NAMEF/2H7R, 2HIS, 2H02/
0274 C DATA IMAPL/2H40, 2H, S, 2HEA, 2H N, 2HAP, 2H /
0275 C 2H40, 2H, L, 2HAN, 2HD, 2HNA, 2HP /
0276 C 2H41, 2H, S, 2HEA, 2H N, 2HAP, 2H /
0277 C 2H41, 2H, L, 2HAN, 2HD, 2HNA, 2HP /
0278 C 2H17, 2H, S, 2HEA, 2H N, 2HAP, 2H /
0279 C 2H17, 2H, L, 2HAN, 2HD, 2HNA, 2HP /
0280 C 2H39, 2H, S, 2HEA, 2H N, 2HAP, 2H /
0281 C 2H39, 2H, L, 2HAN, 2HD, 2HNA, 2HP /
0282 C DATA LCHAR/1HL /
0283 C DATA IX/5450, 5411, 4930, 4925, 8750, 8730, 4100, 4100 /
0284 C DATA IY/2630, 8243, 2465, 8030, 2990, 8600, 1700, 7300 /
0285 C
0286 C IS THIS THE FIRST TIME THROUGH THIS SUBROUTINE? --
0287 C IF NOT, IT IS NOT NECESSARY TO CALCULATE THE INDEX OF THE
0288 C COORDINATES, I, AGAIN
0289 C
0290 C IF(I<GTMP .NE. 0)GO TO 7
0291 C
0292 C THIS IS THE FIRST TIME THROUGH -- READ IN THE COMPLEX NUMBER
0293 C AND THE DESIRED MAP, I.E. SEA OR LAND
0294 C
0295 C CALL PREAD(NAMEF, 7, ILINE)
0296 C
0297 C
0298 C
```

ORIGINAL PAGE IS  
OF POOR QUALITY.

```

0299 CALL LERS(YPOS,LVERSN)
0300 CALL CHAR(0.,YPOS,0,ILINE,64,0,0)
0301 YPOS = YPOS - 16.
0302 IF(LVERSN.EQ.0)GO TO 10
0303 IF(YPOS.LT.48.)YPOS = 458.
0304 IF(IHVHCL.EQ.1) CALL DREAD(NAMEF,9,ILINE)
0305 IF(IHVHCL.EQ.2) CALL DREAD(NAMEF,9,ILINE)
0306 IF(IHVHCL.EQ.0) CALL DREAD(NAMEF,10,ILINE)
0307 CALL LERS(YPOS,LVERSN)
0308 CALL CHAR(24.,YPOS,0,ILINE(2),8,3,0)
0309 CALL CHAR(95.,YPOS,0,ILINE(7),50,0,0)
0310 CALL ITH(I,JTYPE,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.)
0311 CALL CHAR(0.,YPOS,0,TERS,64,0,0)
0312 CALL CHAR(200.,YPOS+16,0,TERS,25,0,0)
0313 IF(IXC.LT.6.AND.IHVHCL.EQ.1) I=1
0314 IF(IXC.GT.5.AND.IXC.LT.12.AND.IHVHCL.EQ.1) I=2
0315 IF(IXC.GT.11.AND.IXC.LT.18.AND.IHVHCL.EQ.1) I=3
0316 IF(IXC.GT.17.AND.IHVHCL.EQ.1) I=4
0317 IF(IXC.LT.6.AND.IHVHCL.EQ.2) I=5
0318 IF(IXC.GE.6.AND.IHVHCL.EQ.2) I=6
0319 IF(IXC.LT.6.AND.IHVHCL.EQ.0) I=7
0320 IF(IXC.GE.6.AND.IHVHCL.EQ.0) I=8
0321 IHP = (I - 1)*6 + 1
0322 CALL CHAR(208.,YPOS+16.,0,IMAPL(IMP),12,0,0)
0323 YPOS = YPOS - 16.
0324 IF(LVERSN.EQ.0)GO TO 6
0325 IF(YPOS.LT.48.)YPOS = 458.
0326 C
0327 C SET THE COORDINATES BASED ON THE INDEX I
0328 C
0329 C*** SET PAD INDEX FOR FUTURE USE...
0330 6 IGOIMP = I
0331 7 CONTINUE
0332 C
0333 C*** RELOAD PAD INDEX PREVIOUSLY COMPUTED
0334 I = IGOIMP
0335 IX0 = IX(I)
0336 IY0 = IY(I)
0337 C
0338 C RETURN TO THE CALLING PROGRAM
0339 C
0340 C RETURN
0341 C
0342 C END OF ORGIN
0343 C
0344 C
0345 C
0346 C
0347 C
0348 C
0349 C
0350 C
0351 C
0352 C
0353 C
0354 C
0355 C
0356 C
0357 C
0358 C
0359 C
0360 C
0361 C
0362 C
0363 C
0364 C
0365 C
0366 C
0367 C
0368 C
0369 C
0370 C
0371 C
0372 C
0373 C
0374 C
0375 C
0376 C
0377 C
0378 C
0379 C
0380 C
0381 C
0382 C
0383 C
0384 C
0385 C
0386 C
0387 C
0388 C
0389 C
0390 C
0391 C
0392 C
0393 C
0394 C
0395 C
0396 C
0397 C
0398 C
0399 C
0400 C
0401 C
0402 C
0403 C
0404 C
0405 C
0406 C
0407 C
0408 C
0409 C
0410 C
0411 C
0412 C
0413 C
0414 C
0415 C
0416 C
0417 C
0418 C
0419 C
0420 C
0421 C
0422 C
0423 C
0424 C
0425 C
0426 C
0427 C
0428 C
0429 C
0430 C
0431 C
0432 C
0433 C
0434 C
0435 C
0436 C
0437 C
0438 C
0439 C
0440 C
0441 C
0442 C
0443 C
0444 C
0445 C
0446 C
0447 C
0448 C
0449 C
0450 C
0451 C
0452 C
0453 C
0454 C
0455 C
0456 C
0457 C
0458 C
0459 C
0460 C
0461 C
0462 C
0463 C
0464 C
0465 C
0466 C
0467 C
0468 C
0469 C
0470 C
0471 C
0472 C
0473 C
0474 C
0475 C
0476 C
0477 C
0478 C
0479 C
0480 C
0481 C
0482 C
0483 C
0484 C
0485 C
0486 C
0487 C
0488 C
0489 C
0490 C
0491 C
0492 C
0493 C
0494 C
0495 C
0496 C
0497 C
0498 C
0499 C
0500 C
0501 C
0502 C
0503 C
0504 C
0505 C
0506 C
0507 C
0508 C
0509 C
0510 C
0511 C
0512 C
0513 C
0514 C
0515 C
0516 C
0517 C
0518 C
0519 C
0520 C
0521 C
0522 C
0523 C
0524 C
0525 C
0526 C
0527 C
0528 C
0529 C
0530 C
0531 C
0532 C
0533 C
0534 C
0535 C
0536 C
0537 C
0538 C
0539 C
0540 C
0541 C
0542 C
0543 C
0544 C
0545 C
0546 C
0547 C
0548 C
0549 C
0550 C
0551 C
0552 C
0553 C
0554 C
0555 C
0556 C
0557 C
0558 C
0559 C
0560 C
0561 C
0562 C
0563 C
0564 C
0565 C
0566 C
0567 C
0568 C
0569 C
0570 C
0571 C
0572 C
0573 C
0574 C
0575 C
0576 C
0577 C
0578 C
0579 C
0580 C
0581 C
0582 C
0583 C
0584 C
0585 C
0586 C
0587 C
0588 C
0589 C
0590 C
0591 C
0592 C
0593 C
0594 C
0595 C
0596 C
0597 C
0598 C
0599 C
0600 C
0601 C
0602 C
0603 C
0604 C
0605 C
0606 C
0607 C
0608 C
0609 C
0610 C
0611 C
0612 C
0613 C
0614 C
0615 C
0616 C
0617 C
0618 C
0619 C
0620 C
0621 C
0622 C
0623 C
0624 C
0625 C
0626 C
0627 C
0628 C
0629 C
0630 C
0631 C
0632 C
0633 C
0634 C
0635 C
0636 C
0637 C
0638 C
0639 C
0640 C
0641 C
0642 C
0643 C
0644 C
0645 C
0646 C
0647 C
0648 C
0649 C
0650 C
0651 C
0652 C
0653 C
0654 C
0655 C
0656 C
0657 C
0658 C
0659 C
0660 C
0661 C
0662 C
0663 C
0664 C
0665 C
0666 C
0667 C
0668 C
0669 C
0670 C
0671 C
0672 C
0673 C
0674 C
0675 C
0676 C
0677 C
0678 C
0679 C
0680 C
0681 C
0682 C
0683 C
0684 C
0685 C
0686 C
0687 C
0688 C
0689 C
0690 C
0691 C
0692 C
0693 C
0694 C
0695 C
0696 C
0697 C
0698 C
0699 C
0700 C
0701 C
0702 C
0703 C
0704 C
0705 C
0706 C
0707 C
0708 C
0709 C
0710 C
0711 C
0712 C
0713 C
0714 C
0715 C
0716 C
0717 C
0718 C
0719 C
0720 C
0721 C
0722 C
0723 C
0724 C
0725 C
0726 C
0727 C
0728 C
0729 C
0730 C
0731 C
0732 C
0733 C
0734 C
0735 C
0736 C
0737 C
0738 C
0739 C
0740 C
0741 C
0742 C
0743 C
0744 C
0745 C
0746 C
0747 C
0748 C
0749 C
0750 C
0751 C
0752 C
0753 C
0754 C
0755 C
0756 C
0757 C
0758 C
0759 C
0760 C
0761 C
0762 C
0763 C
0764 C
0765 C
0766 C
0767 C
0768 C
0769 C
0770 C
0771 C
0772 C
0773 C
0774 C
0775 C
0776 C
0777 C
0778 C
0779 C
0780 C
0781 C
0782 C
0783 C
0784 C
0785 C
0786 C
0787 C
0788 C
0789 C
0790 C
0791 C
0792 C
0793 C
0794 C
0795 C
0796 C
0797 C
0798 C
0799 C
0800 C
0801 C
0802 C
0803 C
0804 C
0805 C
0806 C
0807 C
0808 C
0809 C
0810 C
0811 C
0812 C
0813 C
0814 C
0815 C
0816 C
0817 C
0818 C
0819 C
0820 C
0821 C
0822 C
0823 C
0824 C
0825 C
0826 C
0827 C
0828 C
0829 C
0830 C
0831 C
0832 C
0833 C
0834 C
0835 C
0836 C
0837 C
0838 C
0839 C
0840 C
0841 C
0842 C
0843 C
0844 C
0845 C
0846 C
0847 C
0848 C
0849 C
0850 C
0851 C
0852 C
0853 C
0854 C
0855 C
0856 C
0857 C
0858 C
0859 C
0860 C
0861 C
0862 C
0863 C
0864 C
0865 C
0866 C
0867 C
0868 C
0869 C
0870 C
0871 C
0872 C
0873 C
0874 C
0875 C
0876 C
0877 C
0878 C
0879 C
0880 C
0881 C
0882 C
0883 C
0884 C
0885 C
0886 C
0887 C
0888 C
0889 C
0890 C
0891 C
0892 C
0893 C
0894 C
0895 C
0896 C
0897 C
0898 C
0899 C
0900 C
0901 C
0902 C
0903 C
0904 C
0905 C
0906 C
0907 C
0908 C
0909 C
0910 C
0911 C
0912 C
0913 C
0914 C
0915 C
0916 C
0917 C
0918 C
0919 C
0920 C
0921 C
0922 C
0923 C
0924 C
0925 C
0926 C
0927 C
0928 C
0929 C
0930 C
0931 C
0932 C
0933 C
0934 C
0935 C
0936 C
0937 C
0938 C
0939 C
0940 C
0941 C
0942 C
0943 C
0944 C
0945 C
0946 C
0947 C
0948 C
0949 C
0950 C
0951 C
0952 C
0953 C
0954 C
0955 C
0956 C
0957 C
0958 C
0959 C
0960 C
0961 C
0962 C
0963 C
0964 C
0965 C
0966 C
0967 C
0968 C
0969 C
0970 C
0971 C
0972 C
0973 C
0974 C
0975 C
0976 C
0977 C
0978 C
0979 C
0980 C
0981 C
0982 C
0983 C
0984 C
0985 C
0986 C
0987 C
0988 C
0989 C
0990 C
0991 C
0992 C
0993 C
0994 C
0995 C
0996 C
0997 C
0998 C
0999 C
1000 C

```



ORIGINAL PAGE IS  
OF POOR QUALITY

```

0359 IV=IV
0360 WRITE(LUFLT)-1,-1,IX,IV
0361 RETURN
0362 END
0363 SUBROUTINE DREAD(NAMEF,LNUM,ILINE)
0364 C
0365 C
0366 C
0367 C - THIS SUBROUTINE READS A SPECIFIED QUESTION FROM A SPECIFIED
0368 C - QUESTION FILE. THE QUESTION IS THEN DISPLAYED UPON THE
0369 C - PLASMASCOPE FOR PROCESSING.
0370 C
0371 C
0372 C
0373 DIMENSION NAMEF(3),IDCB(144),IBUF(40),ILINE(32)
0374 CALL OPEN(IDCB,IERR,NAMEF,0)
0375 LOOP = LNUM - 1
0376 DO 10 I=1,LOOP
0377 CALL BLANK(IBUF,40)
0378 CALL READF(IDCB,IERR,IBUF)
0379 CONTINUE
0380 CALL BLANK(IBUF,40)
0381 CALL READF(IDCB,IERR,IBUF)
0382 CALL CODE
0383 READ(IBUF,100) (ILINE(I),I=1,32)
0384 FORMAT(32A2)
0385 CALL CLOSE(IDCB,IERR)
0386 RETURN
0387 END
0388 SUBROUTINE BLANK(IBUF,II)
0389 C
0390 C
0391 C - THIS SUBROUTINE FILLS A SPECIFIED ARRAY WITH A SPECIFIED
0392 C - NUMBER OF BLANKS.
0393 C
0394 C
0395 C
0396 C
0397 DIMENSION IBUF(40)
0398 DATA IBLK/2H /
0399 DO 10 I=1,II
0400 IBUF(I) = IBLK
0401 RETURN
0402 END
0403 SUBROUTINE LERS(YPGS,LVERSH)
0404 C
0405 C
0406 C
0407 C - THIS SUBROUTINE ERASES A SPECIFIED LINE OF 64-CHARACTERS
0408 C - FROM THE PLASMASCOPE.
0409 C
0410 C
0411 C
0412 DIMENSION IERS(32)
0413 DATA IERS/32*2H /
0414 IF(YPGS .GT. 48.0)GO TO 4
0415 IF(LVERSH.EQ.0)GO TO 4
0416 YPGS = 458.0
0417 CALL CLEFZ
0418 RETURN

```

ORIGINAL PAGE IS  
OF POOR QUALITY

4 CALL CHAR(0,0,YPOS,0,TERS,64,0,0)  
CALL CHAR(0,0,YPOS-16,0,0,TERS,64,0,0)  
RETURN  
END  
END\*

0419  
0420  
0421  
0422  
0423

## 4.2 AUXILIARY SUBROUTINE LISTINGS

This section contains the current REEDS auxiliary subroutine listings. They are listed according to the sequence indicated in Section 4.0.



ORIGINAL PAGE IS  
OF POOR QUALITY

```
0459 C      0 - 10'S OF MILLISECDS
0060 C      1 - MILLISECDS
0061 C      2 - SECDS
0062 C      3 - MINUTES
0063 C      IERR - RETURNED ERROR FLAG
0064 C      1 - REQUEST ACCEPTED
0065 C      3 - ILLEGAL PARAMETER
0066 C
0067 C      IF<MULT.GT.0 .AND. IRES.GE.0 .AND. IRES.LE.3>GO TO 4
0068 C      IERR = 3
0069 C      RETURN
0070 C
0071 C      4 IERR = 1
0072 C      IM = MULT
0073 C      IF<IRES.EQ.1>IM = IM/10
0074 C      IR = IRES
0075 C      IF<IRES.EQ.0>IR = 1
0076 C      CALL EXEC<12,0,IR,0,-IM>
0077 C      RETURN
0078 C      END
0079 C
0080 C      SUBROUTINE INK<ITYPE,JTYPE,X,Y,EXY,ICHAR,NCHAR,IXL,IXH,IYL,IYH,
0081 C      COMMON IDUMK<64>,LU,IDUM1<492>
0082 C      100 FORMAT<32A2>
0083 C      101 FORMAT<12>
0084 C      INTEGER UP0,UP9,UP8L
0085 C      DIMENSION ICHAR<1>
0086 C      DATA UP0/030000B/, UP9/034400B/, UPBL/020000B/,
0087 C      LOW0/000060B/, LOW9/000071B/, LOWBL/000040B/,
0088 C      JTYPE = 0
0089 C      IF<ITYPE.EQ.1>GO TO 4
0090 C      INORDS = <NCHAR - 1>/2 + 1
0091 C      READ<LU,100><ICHAR<1>,I=1,INORDS>
0092 C      IF<ITYPE.EQ.0>RETURN
0093 C      IXIN = ICHAR<1>
0094 C      GO TO 7
0095 C
0096 C      4 READ<LU,100> IXIN
0097 C      7 IXUP = IAND<IXIN,177400B>
0098 C      IF<IXUP.EQ.UP8L>IXUP = UP0
0099 C      IXLOW = IAND<IXIN,000377B>
0100 C      IF<IXLOW.NE.LOWBL>GO TO 9
0101 C      IXLOW = I8H<IXUP,8>
0102 C      IXUP = UP0
0103 C
0104 C      9 IF<IXUP.GE.UP0 .AND. IXUP.LE.UP9 .AND.
0105 C      IXLOW.GE.LOW0 .AND. IXLOW.LE.LOW9>GO TO 12
0106 C      IF<ITYPE.EQ.2>RETURN
0107 C      GO TO 4
0108 C
0109 C      12 IXIN = IOR<IXUP,IXLOW>
0110 C      CALL CODE
0111 C      READ<IXIN,101> IX
0112 C      IF<IX.GE.IXL .AND. IX.LE.IXH>GO TO 15
0113 C      IF<ITYPE.EQ.2>RETURN
0114 C      GO TO 4
0115 C
0116 C      15 JTYPE = 1
0117 C      RETURN
0118 C      END
0119 C      SUBROUTINE VERSN<IVER>
0120 C      IVER = 1
0121 C      RETURN
0122 C      END
```

ORIGINAL PAGE IS  
OF POOR QUALITY

4ML162 T-00003 IS ON CR00003 USING 00016 BLKS R=0000

```
0001 FT14,L
0002 SUBROUTINE CHARC(X,Y,IXY,ICHAR,NCMAP,MODE,ISIZE)
0003 REEDS PLOTTER GRAPHICS
0004 COMMON IDUMK(64),LU,IDUM1(492)
0005 DIMENSION ICHAR(32),JCHAR(33)
0006 IF(MODE.EQ.1)RETURN
0007 JCHAR(1) = ICHAR
0008 DO 5 K=2,33
0009 JCHAR(K) = ICHAR(K-1)
0010 XX = X/51.2
0011 YY = Y/51.2
0012 SIZE = 10./64
0013 IF(SIZE.EQ.1) SIZE = SIZE/1.5
0014 CALL SYMB(XX,YY,SIZE,JCHAR,0.,1)
0015 RETURN
0016 END
0017 SUBROUTINE LINE(X,Y,MXY,MODE)
0018 DIMENSION X(1),Y(1)
0019 IF(MXY.LT.2.OR.MODE.NE.0)RETURN
0020 XX = X(1)/51.2
0021 YY = Y(1)/51.2
0022 CALL PLOT(XX,YY,3)
0023 DO 7 I=2,MXY
0024 XX = X(I)/51.2
0025 YY = Y(I)/51.2
0026 CALL PLOT(XX,YY,2)
0027 RETURN
0028 END
0029 SUBROUTINE GRAF(IL)
0030 CALL PLTLU(12)
0031 CALL LLEFT
0032 RETURN
0033 END
0034 SUBROUTINE MGRAF
0035 CALL URITE
0036 RETURN
0037 END
0038 SUBROUTINE CLEAR
0039 COMMON IDUMK(64),LU,IDUM1(492)
0040 C EY
0041 200 FORMAT ('E')
0042 C EZ
0043 WRITE (LU,200)
0044 CALL DELAY(0.1)
0045 RETURN
0046 END
0047 SUBROUTINE LALT(I1,I2,I3)
0048 RETURN
0049 END
0050 SUBROUTINE CHARC(X,Y,IXY,ICHAR,NCMAP,MODE,INT)
0051 COMMON IDUMK(64),LU,IDUM1(492)
0052 C EY
0053 200 FORMAT ('E',14"14"R"32A2)
0054 201 FORMAT ('E',14"14"R"14"R"32A2)
0055 C EZ
0056 INTEGER BLANKS,CHOUT(32)
0057 DIMENSION ICHAR(1)
0058 IN 6 S/21
```

```

0059 IF<IXY .GE. 0>GO TO 2
0060 IX = INT<X>
0061 IY = 31 - INT<Y>
0062 GO TO 4
0063 2 IX = INT<0.125 * X>
0064 IY = 31 - INT<0.0625 * Y>
0065 4 IF<IY .LE. 22>GO TO 7
0066 IY = IY - 22
0067 GO TO 4
0068 7 NWORDS = (NCHAR - 1)/2 + 1
0069 IF<MODE .NE. 1>GO TO 15
0070 DO 12 I=1,NWORDS
0071 12 CHOUT(I) = BLANKS
0072 GO TO 19
0073 15 DO 16 I=1,NWORDS
0074 16 CHOUT(I) = ICHAR(I)
0075 19 IF<MODE .EQ. 3>GO TO 25
0076 WRITE (LU,200) IX,IY,<CHOUT(I),I=1,NWORDS>
0077 GO TO 28
0078 25 WRITE (LU,201) IX,IY,<CHOUT(I),I=1,NWORDS>
0079 28 CALL DELAY<0.1>
0080 RETURN
0081 END
0082 SUBROUTINE DELAY<SEC>
0083 MSEC = 1000.0 * SEC
0084 CALL WAIT<MSEC,1>
0085 RETURN
0086 END
0087 SUBROUTINE WAIT<MULT,IRES,IERR>
0088 C
0089 C THIS ROUTINE DUPLICATES THE FUNCTION OF THE WAIT SUBROUTINE
0090 C FOUND IN THE ISA LIBRARY. THE INPUT VARIABLES ARE DEFINED
0091 C AS FOLLOWS:
0092 C MULT - POSITIVE INTEGER INDICATING THE NUMBER OF UNITS
0093 C THE CALLING PROGRAM SHOULD BE PUT IN A WAIT STATE
0094 C IRES - THE RESOLUTION OF MULT, I.E.
0095 C 0 - 10'S OF MILLISECONDS
0096 C 1 - MILLISECONDS
0097 C 2 - SECONDS
0098 C 3 - MINUTES
0099 C IERR - RETURNED ERROR FLAG
0100 C 1 - REQUEST ACCEPTED
0101 C 3 - ILLEGAL PARAMETER
0102 C
0103 IF<MULT.GT.0 .AND. IRES.GE.0 .AND. IRES.LE.3>GO TO 4
0104 IERR = 3
0105 RETURN
0106 4 IERR = 1
0107 IM = MULT
0108 IF<IRES .EQ. 1>IM = IM/10
0109 IR = IRES
0110 IF<IRES .EQ. 0>IR = 1
0111 CALL EXEC<12,0,IR,0,-IM>
0112 RETURN
0113 END
0114 SUBROUTINE INK<ITYPE,JTYPE,X,Y,IXY,ICOMP,NCHAR,IXL,IXM,IYL,IYM,
0115 IX,IY>
0116 COMMON IDUM<64>,LU,JDUM<32>
0117 FORMAT (32A2)
0118 101 FORMAT (12)

```

```
0119 INTEGER UP0,UP9,UP6L
0120 DIMENSION ICHAR(1)
0121 DATA UP0/'030000B', UP9/'034400B', UP6L/'020000B',
0122 LOW0/'000000B', LOW9/'000071B', LOW6L/'000040B',
0123 JTYPE = 0
0124 IF(IJTYPE .EQ. 1)GO TO 4
0125 IMORDS = ICHAR - 1)/2 + 1
0126 READ (LU,100) (ICAR(I),I=1,IMORDS)
0127 IF(IJTYPE .EQ. 0)RETURN
0128 IXIN = ICHAR(1)
0129 GO TO 7
0130 4 READ (LU,100) IXIN
0131 7 IXUP = IAND(IXIN,177400B)
0132 IF(IXUP .EQ. UP6L)IXUP = UP0
0133 IXLOW = IAND(IXIN,000377B)
0134 IF(IXLOW .NE. LOW6L)GO TO 9
0135 IXLOW = ISHIF(IXUP,8)
0136 IXUP = UP0
0137 9 IF(IXUP.GE.UP0 .AND. IXUP.LE.UP9 .AND.
0138 IXLOW.GE.LOW0 .AND. IXLOW.LE.LOW9)GO TO 12
0139 IF(IJTYPE .EQ. 2)RETURN
0140 GO TO 4
0141 12 IXIN = IOR(IXUP,IXLOW)
0142 CALL CODE
0143 READ (IXIN,101) IX
0144 IF(IX.GE.IXL .AND. IX.LE.IXH)GO TO 15
0145 IF(IJTYPE .EQ. 2)RETURN
0146 GO TO 4
0147 15 JTYPE = 1
0148 RETURN
0149 END
0150 SUBROUTINE VERSK(IVER)
0151 IVER = 1
0152 RETURN
0153 END
0154 END*
```





ORIGINAL PAGE IS  
OF POOR QUALITY

0059	*				
0060		OLD XTMP, I	STORE XORG		
0061		DST XORG			
0062	*				
0063		OLD YTMP, I	STORE YORG		
0064		DST YORG			
0065	*				
0066		JMP SETOR, I	RETURN TO CALLING PROGRAM		
0067	*				
0068	*				
0069	*				
0070	*				
0071	*	SETSC NOP	ENTRY/EXIT LINE		
0072	*				
0073		JSB .ENTR	GET ADDRESSES OF XSCAL, YSCAL		
0074		DEF XTMP			
0075	*				
0076		OLD XTMP, I	STORE XSCAL		
0077		DST XSCAL			
0078	*				
0079		OLD YTMP, I	STORE YSCAL		
0080		DST YSCAL			
0081	*				
0082	*	JMP SETSC, I	RETURN TO CALLING PROGRAM		
0083	*				
0084	*				
0085	*				
0086	*				
0087	*	GETOR NOP	ENTRY/EXIT LINE		
0088	*				
0089	*	JSB .ENTR	GET ADDRESSES OF XORG, YORG		
0090	*	DEF XTMP			
0091	*				
0092		OLD XORG	RETURN XORG AND REPLACE WITH 0		
0093		DST XTMP, I			
0094		OLD FLT0			
0095		DST XORG			
0096	*				
0097		OLD YORG	RETURN YORG AND REPLACE WITH 0		
0098		DST YTMP, I			
0099		OLD FLT0			
0100		DST YORG			
0101	*				
0102	*	JMP GETOR, I	RETURN TO CALLING PROGRAM		
0103	*				
0104	*				
0105	*				
0106	*				
0107	*	SETSC NOP	ENTRY EXIT LINE		
0108	*				
0109	*	JSB .ENTR	GET ADDRESSES OF XSCAL, YSCAL		
0110	*	DEF XTMP			
0111	*				
0112		OLD XSCAL	RETURN XSCAL AND REPLACE WITH I		
0113		DST XTMP, I			
0114		OLD FLT1			
0115		DST XSCAL			
0116	*				
0117	*	OLD YSCAL	RETURN YSCAL AND REPLACE WITH I		
0118	*	... YTH...			

ORIGINAL PAGE IS  
OF POOR QUALITY

```
0113 OLD FLTI
0120 DST YSCAL
0121 *
0122 JMP GETSC,I RETURN TO CALLING PROGRAM
0123 *
0124 *
0125 *
0126 *
0127 LASTH NOP LAST MAKE WORD OUTPUT
0128 *
0129 *
0130 *
0131 *
0132 P15 DEC 0.5
0133 FLT6 DEC 0.0
0134 FLT1 DEC 1.0
0135 *
0136 XORG DEC 0.0
0137 YORG DEC 0.0
0138 *
0139 XSCAL DEC 1.0
0140 YSCAL DEC 1.0
0141 *
0142 *
0143 *
0144 *
0145 END CALIX
```

ORIGINAL PAGE IS  
OF POOR QUALITY

MDLINE T=00003 IS ON CR00003 USING 00025 BLKS K=0000

```

0001 ASMB,R,F,I
0002 *
0003 * THIS ROUTINE PLOTS DASHED LINES ON THE PLASMASCOPE
0004 * WITH THE LENGTH OF THE DASHES AND THE LENGTH OF THE
0005 * SPACES VARIABLE
0006 *
0007 * NAM DLIN,7 REEDS GRAPHICS
0008 * ENT DLIN
0009 * EXT .ENTR,GETOR,GETSC, SORT,LIN,SETOR,SETSC
0010 *
0011 X BSS I
0012 Y BSS I
0013 HXY BSS I
0014 MODE BSS I
0015 LDASH BSS I
0016 LSFC BSS I
0017 *
0018 DLIN NOP ENTRY/EXIT LINE
0019 *
0020 JSB .ENTR GET ADDRESSES OF X,Y,HXY,ETC.
0021 DEF X
0022 *
0023 LDA HXY,I IS HXY < 2 ?
0024 ADA NEG1
0025 CMA,INA
0026 SSA,R8S
0027 JMP DLIN,I YES -- RETURN TO CALLING PROGRAM
0028 STA NUM NO -- SET POINT COUNTER
0029 *
0030 JSB GETOR GET ORIGIN VALUES AND SET THE ORIGIN
0031 DEF ++J VALUES EQUAL TO 0 FOR CALLS TO THE
0032 DEF XORG EXTERNAL ROUTINES CALIX AND CALY
0033 DEF YORG
0034 *
0035 JSB GETSC GET SCALE VALUES AND SET THE SCALE
0036 DEF ++J VALUES EQUAL TO 1 FOR CALLS TO THE
0037 DEF XSCAL EXTERNAL ROUTINES CALIX AND CALY
0038 DEF YSCAL
0039 *
0040 LDA LDASH,I STORE DASH LENGTH AND SPACE
0041 STA LENS LENGTH IN LENS ARRAY
0042 LDA LSFC,I
0043 STA LENS+I
0044 *
0045 CLA CLEAR LEFTOVER LENGTH AND THE
0046 STA LOLEN DASH OR SPACE INDICATOR
0047 STA DORS
0048 *
0049 LDA MODE,I STORE THE MODE
0050 STA MODED
0051 *
0052 LDA X,I GET, CONVERT TO RASTER UNITS,
0053 ISZ X AND STORE THE FIRST POINT ON
0054 LOB X,I THE LINE AS POINT I
0055 ISZ X
0056 JSB CALIX
0057 *
0058 XX
0059

```

ORIGINAL PAGE IS  
OF POOR QUALITY

0659	LDA Y, I	
0660	ISZ Y	
0661	LDB Y, I	
0662	ISZ Y	
0663	JSB CALIY	
0664	DST YY	
0665		
0666	LDA X, I	GET, CONVERT TO RASTER UNITS,
0667	ISZ X	AND STORE THE NEXT POINT ON
0668	LDB X, I	THE LINE AS POINT 2
0669	ISZ X	
0670	JSB CALIX	
0671	DST XNEXT	
0672		
0673	LDA Y, I	
0674	ISZ Y	
0675	LDB Y, I	
0676	ISZ Y	
0677	JSB CALIY	
0678	DST YNEXT	
0679		
0680	FSB YY	DETERMINE THE LENGTH FROM
0681	FMP A	POINT 1 TO POINT 2
0682	DST TEND	
0683	OLD XNEXT	
0684	FSB XX	
0685	FMP A	
0686	FAD TEMP	
0687	JSB SORT	
0688	NOP	
0689	FAD PTS	INTEGERIZE AND STORE IN ALEN
0690	FIX	
0691	STA ALEN	
0692	FLT TEMP	FLOAT AND STORE IN TEMP
0693	DST TEMP	
0694		
0695	LDA ALEN	ADD IN ANY LEFTOVER LENGTH FROM
0696	ADA LOLEN	THE LAST TIME AND STORE IN LOLEN
0697	STA LOLEN	
0698		
0699	CMA, IMA	COMPARE WITH DESIRED LENGTH OF
0700	LEY DORS	DASH OR SPACE
0701	LBY LENS	
0702	ADA B	
0703	SSA	
0704	JMP TOFAR	POINTS TOO FAR APART
0705	STA TEMP	
0706		
0707	OLD XNEXT	POINTS TOO CLOSE TOGETHER OR EXACT
0708	DST XX+2	
0709	OLD YNEXT	
0710	DST YY+2	
0711		
0712	JSB LINE	OUTPUT POINT 1 AND POINT 2
0713	DEF **5	
0714	DEF XX	
0715	DEF YY	
0716	DEF TMD	
0717	DEF MOED	
0718		

ORIGINAL PAGE IS  
OF POOR QUALITY

0119	LDA TEMP	IF POINTS EXACT -- CHANGE INDICATORS
0120	SZA, RSS	
0121	JSB CHANG	
0122		
0123	DLD XX+2	MOVE POINT 2 INTO POINT 1
0124	DST XX	
0125	DLD YY+2	
0126	DST YY	
0127		
0128	ISZ NUM	ANY MORE POINTS?
0129	JMP NEXT	YES -- GET ANOTHER
0130	JSB SETOR	NO -- RESET THE ORIGIN FOR THE EXTERNAL
0131	DEF **3	CALLS TO CALIX AND CALIY
0132	DEF XORG	
0133	DEF YORG	
0134		
0135	JSB SETSC	RESET THE SCALE FOR THE EXTERNAL
0136	DEF **3	CALLS TO CALIX AND CALIY
0137	DEF XSCAL	
0138	DEF YSCAL	
0139		
0140	JMP DLINE, I	RETURN TO CALLING PROGRAM
0141		
0142	TUFAR ADA ALEN	POINTS TOO FAR APART --
0143	FLT TEMP	CALCULATE AN INTERMEDIATE POINT
0144	FDV TEMP	AND STORE IT
0145	DST TEMP	
0146		
0147	DLD XNEXT	
0148	FSB XX	
0149	FMP TEMP	
0150	FAD XX	
0151	FAD PTS	
0152	FIX	
0153	FLT	
0154	DST XX+2	
0155		
0156	DLD YNEXT	
0157	FSB YY	
0158	FMP TEMP	
0159	FAD YY	
0160	FAD PTS	
0161	FIX	
0162	FLT	
0163	DST YY+2	
0164		
0165	JSB LINE	OUTPUT POINT 1 AND THE
0166	DEF **5	INTERMEDIATE POINT
0167	DEF XX	
0168	DEF YY	
0169	DEF TWO	
0170	DEF MODED	
0171		
0172	JSB CHANG	CHANGE INDICATORS
0173		
0174	DLD XX+2	MOVE INTERMEDIATE POINT INTO
0175	DST XX	POINT 1 -- POINT 2 REMAINS THE
0176	DLD YY+2	SAME
0177	DST YY	
017		

ORIGINAL PAGE IS  
OF POOR QUALITY

0179	OLD YNEXT	TRY AGAIN TO GET TO POINT 2
0180	JMP SAME	
0181	•	
0182	•	
0183	•	
0184	•	
0185	CHANG NOP	SUBPROGRAM TO CHANGE INDICATORS
0186	•	
0187	LDA MODED	SWITCH THE MODE FROM WRITE TO
0188	CMA, INA	ERASE OR SPACE-VEPSA
0189	INA	
0190	STA MODED	
0191	•	
0192	LDA DORS	SWITCH THE DASH OR SPACE INDICATOR
0193	CMA, INA	
0194	INA	
0195	STA DORS	
0196	•	
0197	CLA	CLEAR THE LEFTOVER LENGTH
0198	STA LOLEN	
0199	•	
0200	JMP CHANG, I	RETURN TO THE APPROPRIATE PLACE
0201	•	
0202	•	
0203	•	
0204	•	
0205	CALIX NOP	SUBPROGRAM TO CONVERT AN X
0206	FSB XORG	COORDINATE TO RASTER UNITS BY
0207	FMP XSCAL	X = AINT((X - XORG) * XSCAL + 0.5)
0208	FAD PTS	(SIMILAR TO EXTERNAL CALIX USED
0209	FIX	BY ALL OTHER ROUTINES)
0210	FLT	
0211	JMP CALIX, I	
0212	•	
0213	•	
0214	•	
0215	•	
0216	CALY NOP	SUBPROGRAM TO CONVERT A Y
0217	FSB YORG	COORDINATE TO RASTER UNITS BY
0218	FMP YSCAL	Y = AINT((Y - YORG) * YSCAL + 0.5)
0219	FAD PTS	(SIMILAR TO EXTERNAL CALY USED
0220	FIX	BY ALL OTHER ROUTINES)
0221	FLT	
0222	JMP CALY, I	
0223	•	
0224	•	
0225	•	
0226	•	
0227	A	EQU 0
0228	B	EQU 1
0229	•	
0230	HEG1	DEC -1
0231	TWG	DEC 2
0232	PTS	DEC 0.5
0233	•	
0234	XORG	BSS 2
0235	YORG	BSS 2
0236	•	
0237	XSCAL	BSS 2
0238	YSCAL	BSS 2

ORIGINAL PAGE IS  
OF POOR QUALITY

0239 \*  
0240 HUM NQP  
0241 LOLEH NQP  
0242 DOPS NQP  
0243 MUFED NQP  
0244 ALEH NQP  
0245 \*  
0246 LEHS BSS 2  
0247 \*  
0248 XX BSS 4  
0249 YF BSS 4  
0250 \*  
0251 XNEXE BSS 2  
0252 YNEXE BSS 2  
0253 \*  
0254 TEMP BSS 2  
0255 \*  
0256 END DLIME



#### 4.3 MAIN MODULE QUESTION FILES

This section contains the Listings of the Main Module Question Files. They are listed according to the sequence indicated in Section 4.0



ORIGINAL PAGE IS  
OF POOR QUALITY

```

?REDS T=00003 IS ON CR00031 USING 00006 BLKS R=0000
0001 123456789012345678901234567890123456789012345678901234
0002 *****NASA/MSFC EXPERIMENTAL MULTILAYER DIFFUSION MODEL*****
0003 USE DEFAULT COMMON DATA FILE (DATA01) ? YES NO
0004 Enter Number of Runs and Common Data File (e.g. 04, DATA02)
0005 Keyboard Entry:
0006 Run Will Use Data File
0007 Run Type: RESEARCH-R OPERATIONAL PRODUCTION-P NO
0008 Default Launch Time/Date? YES
0009 Enter Launch Time/Date:
0010 Select Launch Vehicle: DELTA=0 TITAN=T SHUTTLE=S
0011 Enter Top of Surface Layer(M):
0012 Enter Sigma of Wind Azimuth Angle (xx.x):
0013 *****PROGRAM REEDS HAS TERMINATED NORMALLY *****
0014 Desired Processing ? TERMINATE=T RESTART=R
0015 Do You Want Concentrations At:
0016 SURFACE=SU STABILIZATION=ST SOMETHING ELSE=SE
0017 Use Default Height of Calculation? Zz= YES=25 NO=30
0018 Enter Zz:
0019 Use Default Surface Reflection ? Rf=1.0 YES=25 NO=30
0020 Rf= .8 .7 .5 .3 .1 .0 Enter:
0021 Use Default Height of Base Layer ? Zb=0.0 YES=25 NO=30
0022 Enter Zb:
0023 MET Plots? YES NO
0024 Generate The MET Plot Form? YES NO
0025 Generate Printer Output? YES NO
0026 Enter Specific Sounding (i.e. 06, JAN, 5152)
0027 Sounding 01:
0028 Sounding 02:
0029 Sounding 03:
0030 Sounding 04:
0031 Sounding 05:

```

ORIGINAL PAGE IS  
OF POOR QUALITY

2RCLDS T=0003 IS ON CK0003 USING 0002 BLKS R=0000  
0001 123456789012345678901234567890123456789012345678901234  
0002 Estimated Top of Surface Layer(M) is:  
0003 Use: EST=05 MET PROFILE TO COMPUTE=10 ENTER VALUE:  
0004 Enter Desired Top of Surface Layer(M):  
0005 Calculated Top of Surface Transport Layer is: ? YES NO  
0006 Use a Sigma for Wind Azimuth Angle of:  
0007 Enter Desired Sigma of Wind Azimuth Angle (xx.x):

ORIGINAL PAGE IS  
OF POOR QUALITY

```

7RCUNS T=00003 IS ON CR00003 USING 00003 BLKS R=0000
0001 12345678901234567890123456789012345678901234
0002 Centerline Concentration Plot Desired? YES NO
0003 Specific Site Predictions (Plotted)? YES NO
0004 Concentration Isopleth Plot Desired? YES NO
0005 Enter Range(m) and Azimuth(deg): (0 Terminates Procedure)
0006 Range: Azimuth:
0007 Enter Desired Complex: (Sea or Land Map)
0008 40,8=05 40,L=08 41,9=15 41,L=18
0009 17,9=05 17,L=08
0010 39,5=05 39,L=08

```

ORIGINAL PAGE IS  
OF POOR QUALITY

ZRISOS T=00003 IS ON CR00003 USING 00002 R:KS R=0000

0001 123456789012345678901234567890123456789012345678901234  
0002 Default Isopleth Concentration Values are: Conc(1)=  
0003 Conc(2)= Conc(3)= Conc(4)= -1.0  
0004 Use Default Isopleth Concentration Values ? YES NO  
0005 Enter Isopleth Values: (Negative Value Terminates Procedure)  
0006 Enter: Conc( )=  
0007 Enter Desired Complex: (Sea or Land Map)  
0008 40.9=05 40.L=08 41.9=15 41.L=20  
0009 17.8=05 17.L=08  
0010 39.9=05 39.L=08

#### 4.4 EXAMPLE METEOROLOGICAL SOUNDING FILES

This section contains the current listings of two (2) example meteorological sounding files. They are listed according to the sequence indicated in Section 4.0.





ORIGINAL PAGE IS  
OF POOR QUALITY

DATA01 T=00004 IS ON CR00037 USING 00008 BLKS P=0036

0001	TEST NBR	03718	0	MINUS	40	MI
0002	RANTHSONDE RUN	AP/GK	-1			
0003	CAPE CANAVERAL AFS,	FLORIDA				
0004	06312	10	DEC	1974		
0005	ASCERT NBR	0759				
0006	ALT	DIR	SPD	TEMP	D/PRT	PRESS
0007	FT	DEG	KTS	DEG	C	MBS
0008	000016	310	006	07.9	92.5	1023.00
0009	001000	345	021	08.9	03.3	0586.64
0010	002000	355	020	07.7	-0.2	0950.91
0011	003000	346	020	07.3	-4.7	0916.53
0012	004000	332	023	07.1	-5.7	0803.27
0013	005000	314	025	06.5	-7.1	0851.17
0014	006000	306	027	05.8	-7.0	0820.13
0015	007000	319	027	03.6	-8.6	0796.09
0016	008000	301	033	02.6	-10.5	0768.90
0017	009000	311	036	01.6	-11.1	0732.84
0018	010000	317	043	00.2	-11.9	0705.62
0019	TERMINATION					
0020	TRG/PAUSE			22682	FT	6914
0021	MANUATORY LEVELS			0	FEET	0
0022	ALT	DIR	KTS	TEMP	D/PRT	PRESS
0023	000633	339	019	09.1	03.1	1000.0
0024	002023	355	020	07.8	-0.5	0950.0
0025	003487	340	022	07.4	-5.2	0900.0
0026	005029	314	025	06.4	-7.2	0850.0
0027	006655	310	027	04.3	-9.1	0800.0
0028	008371	297	035	02.1	-10.5	0750.6
0029	SIGNIFICANT LEVELS					
0030	ALT	DIR	KTS	TEMP	D/PRT	PRESS
0031	000743	341	019	09.5	03.5	996.00
0032	001827	355	020	06.7	02.2	0957.00
0033	002169	354	020	08.6	-2.4	0945.00
0034	006136	305	027	05.8	-7.9	0916.00
0035	007343	310	030	03.0	-10.4	0860.00
0036	MINIM					